

2018.3 强网杯 部分writeup

转载

[weixin_30765475](#) 于 2018-03-28 10:32:00 发布 69 收藏

文章标签: [php](#)

原文链接: <http://www.cnblogs.com/P201521410043/p/8662354.html>

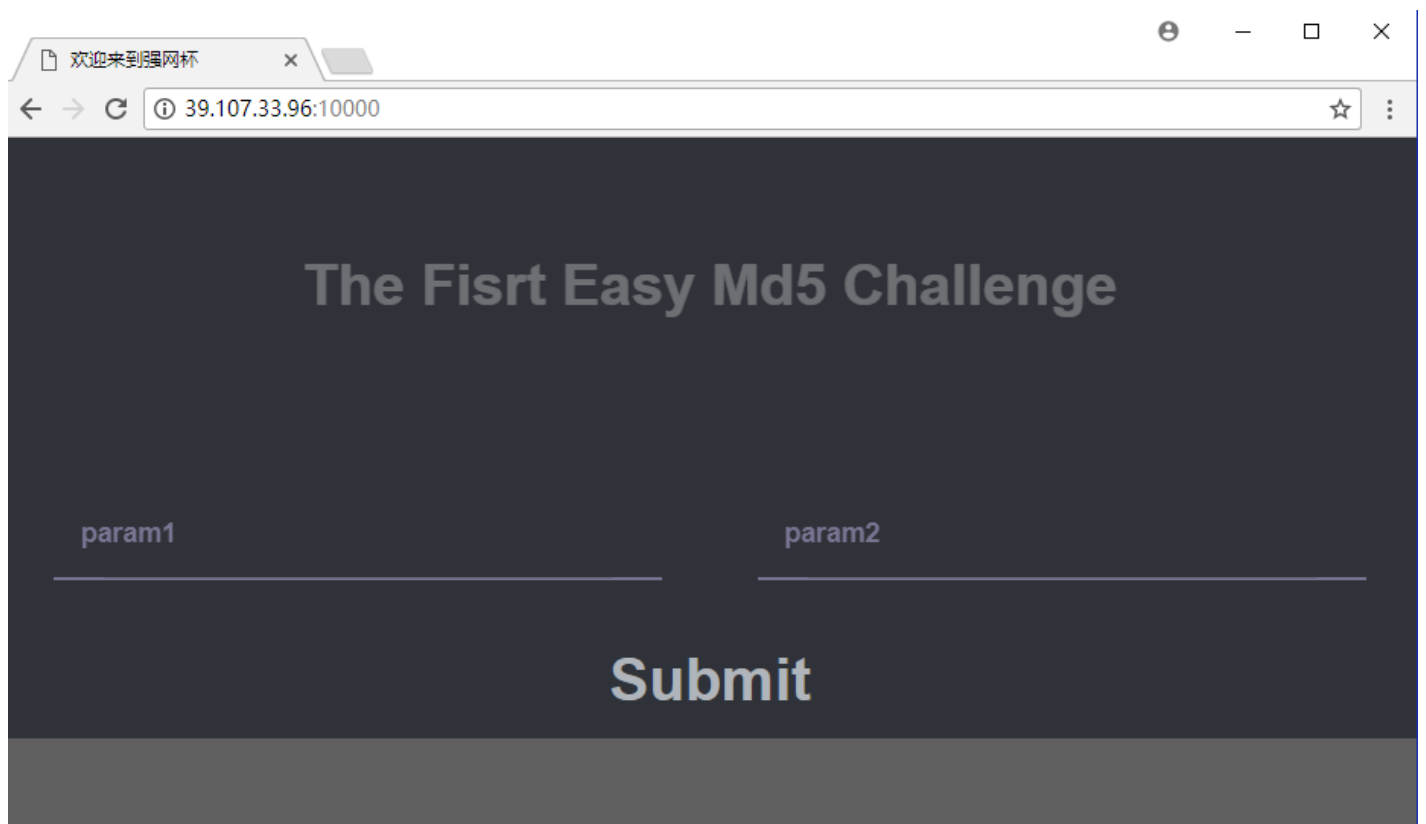
版权

第一次跟大佬打ctf 啥都不会 腆着脸抄了一个writeup

web签到题

原题链接: <http://39.107.33.96:10000/>

emmmm.....



拿到此题，第一反应是MD5碰撞（生日攻击？）

查看网页源码

```

1 <!DOCTYPE html>
2 <html lang="zh" class="no-js">
3   <head>
4     <meta charset="UTF-8" />
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <title>欢迎来到强网杯</title>
8     <link rel="stylesheet" type="text/css" href="css/normalize.css" />
9     <link rel="stylesheet" type="text/css" href="fonts/font-awesome-4.2.0/css/font-awesome.min.css" />
10    <link rel="stylesheet" type="text/css" href="css/demo.css" />
11    <link rel="stylesheet" type="text/css" href="css/component.css" />
12    <script src="js/jquery.min.js"></script>
13  </head>
14  <body>
15    <div class="container">
16      <section class="content bgcolor-4">
17        <h2>The First Easy Md5 Challenge</h2>
18        <!--
19          if($_POST['param1']!= $_POST['param2'] && md5($_POST['param1'])==md5($_POST['param2'])) {
20            die("success!");
21          }
22        -->
23        <span class="input input--madoka">
24          <input class="input_field input_field--madoka" type="text" id="input-31" name='param1' />
25          <label class="input_label input_label--madoka" for="input-31">
26            <svg class="graphic graphic--madoka" width="100%" height="100%" viewBox="0 0 404 77"
preserveAspectRatio="none">
27              <path d="m0,01404,010,771-404,010,-77z"/>
28            </svg>
29            <span class="input_label-content input_label-content--madoka">param1</span>
30          </label>
31        </span>
32        <span class="input input--madoka">
33          <input class="input_field input_field--madoka" type="text" id="input-32" name='param2' />
34          <label class="input_label input_label--madoka" for="input-32">
35            <svg class="graphic graphic--madoka" width="100%" height="100%" viewBox="0 0 404 77"
preserveAspectRatio="none">

```

这里面要求POST上去的参数 param1 != param2 && md5('param1') == md5('param2')

这道题运用了php的一个哈希比较缺陷，就是php在处理0e开头md5哈希字符串时，会将他看成0，PHP在处理哈希字符串时，会利用"!="或"=="来对哈希值进行比较，它把每一个以"0E"开头的哈希值都解释为0，所以如果两个不同的密码经过哈希以后，其哈希值都是以"0E"开头的，那么PHP将会认为他们相同，都是0。

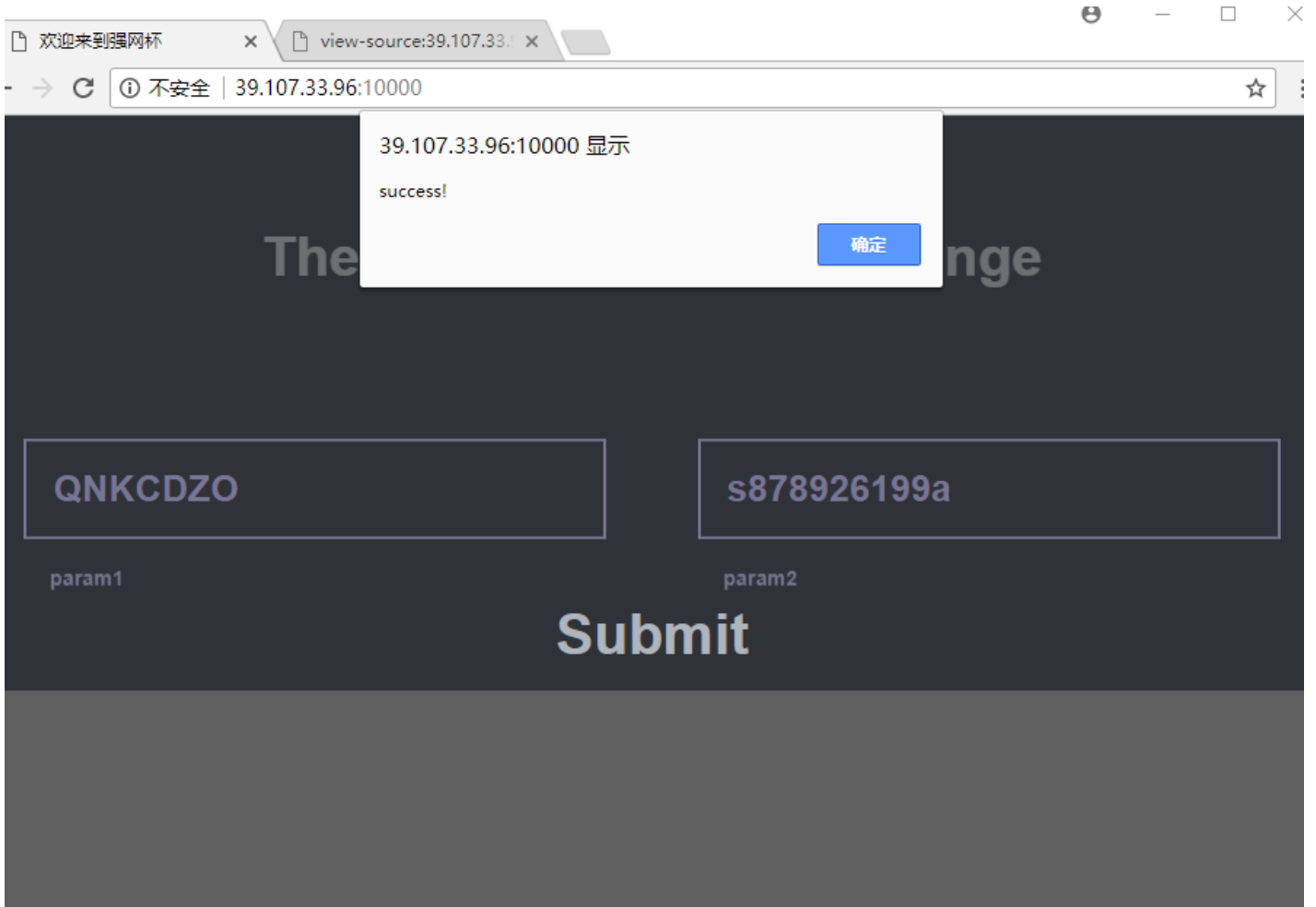
```

C:\Users\n551\Desktop\md5.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
test.php x test1.php x md5.py x 1.txt x 2.txt x md5(2).py x
1 import hashlib
2 md5=hashlib.md5('QNKCDZO'.encode()).hexdigest()
3 print(md5)
4 md5=hashlib.md5('s878926199a'.encode()).hexdigest()
5 print(md5)

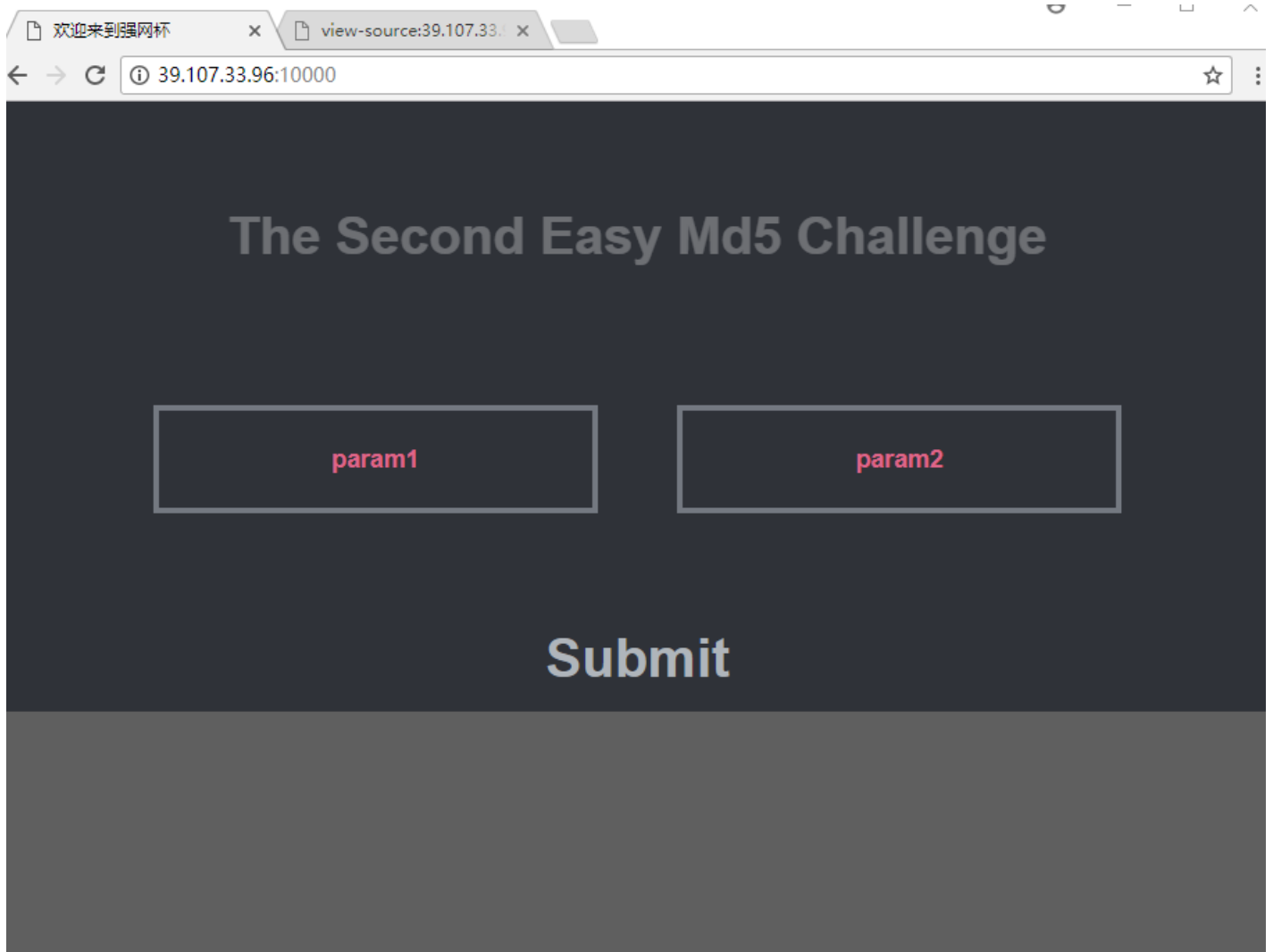
0e830400451993494058024219903391
0e545993274517709034328855841020
[Finished in 2.1s]
Line 5, Column 11 Tab Size: 4 Python

```

提交上去



然后到了第二个页面



看一下源码

```
2     <script src="js/jquery.min.js"></script>
3 </head>
4 <body>
5     <div class="container">
6         <section class="content bgcolor-4">
7             <h2>The Second Easy Md5 Challenge</h2>
8             <!--
9                 if($_POST['param1']!= $_POST['param2'] && md5($_POST['param1'])==md5($_POST['param2'])) {
10                     die("success!");
11                 }
12             -->
13             <span class="input input-kuro">
14                 <input class="input_field input_field-kuro" type="text" id="input-7" name="param1"/>
15                 <label class="input_label input_label-kuro" for="input-7">
16                     <span class="input_label-content input_label-content-kuro">param1</span>
17                 </label>
18             </span>
```

<code>\$a == \$b</code>	Equal	TRUE if <code>\$a</code> is equal to <code>\$b</code> after type juggling.
<code>\$a === \$b</code>	Identical	TRUE if <code>\$a</code> is equal to <code>\$b</code> , and they are of the same type.
<code>\$a != \$b</code>	Not equal	TRUE if <code>\$a</code> is not equal to <code>\$b</code> after type juggling.
<code>\$a <> \$b</code>	Not equal	TRUE if <code>\$a</code> is not equal to <code>\$b</code> after type juggling.
<code>\$a !== \$b</code>	Not identical	TRUE if <code>\$a</code> is not equal to <code>\$b</code> , or they are not of the same type.
<code>\$a < \$b</code>	Less than	TRUE if <code>\$a</code> is strictly less than <code>\$b</code> .
<code>\$a > \$b</code>	Greater than	TRUE if <code>\$a</code> is strictly greater than <code>\$b</code> .
<code>\$a <= \$b</code>	Less than or equal to	TRUE if <code>\$a</code> is less than or equal to <code>\$b</code> .
<code>\$a >= \$b</code>	Greater than or equal to	TRUE if <code>\$a</code> is greater than or equal to <code>\$b</code> .

共3张图,当前是第2张

定义和用法

`md5()` 函数计算字符串的 MD5 散列。

`md5()` 函数使用 RSA 数据安全，包括 MD5 报文摘要算法。

来自 RFC 1321 的解释 - MD5 报文摘要算法：MD5 报文摘要算法将任意长度的信息作为输入值，并将其换算成一个 128 位长度的“指纹信息”或“报文摘要”值来代表这个输入值，并以换算后的值作为结果。MD5 算法主要是为数字签名应用程序而设计的；在这个数字签名应用程序中，较大的文件将在加密（这里的加密过程是通过在一个密码系统下[如：RSA]的公开密钥下设置私有密钥而完成的）之前以一种安全的方式进行压缩。

如需计算文件的 MD5 散列，请使用 `md5_file()` 函数。

语法

```
md5 (string, raw)
```

参数	描述
<code>string</code>	必需。规定要计算的字符串。
<code>raw</code>	可选。规定十六进制或二进制输出格式： <ul style="list-style-type: none"> TRUE - 原始 16 字符二进制格式 FALSE - 默认。32 字符十六进制数

此时比较符已经变成了 `===` 这样的类型

php中，MD5函数所传入的参数必须是字符串类型，那么假设我们传入的不是一个字符串呢？

我们构造一个数组试试

```
1 <?php
2 $a=array("a");
3 $b=array();
4 var_dump(md5($a)==md5($b));
5 var_dump($a==$b);
6 if ($a!=$b&& md5($a)===md5($b))
7 {
8     die("success!");# code...
9 }
10 ?>
```

Warning: md5() expects parameter 1 to be string, array given in E:\AppServ\www\test1.php on line 4

Warning: md5() expects parameter 1 to be string, array given in E:\AppServ\www\test1.php on line 4
bool(true) bool(false)

Warning: md5() expects parameter 1 to be string, array given in E:\AppServ\www\test1.php on line 6

Warning: md5() expects parameter 1 to be string, array given in E:\AppServ\www\test1.php on line 6
success!

可以看出 发现报错了

这里边我们构造了一个数组，传入到md5()这个函数里边，报错提示md5()第一个参数必须为str类型。虽然说if条件不符合，但是程序依然会继续运行。

抓包试一下，把参数改为数组

1 x 2 x 3 x 4 x 5 x 6 x 7 x ...

Go Cancel < >

Target: <http://39.107.33.96:10000>

Request

Raw Params Headers Hex

```
OST / HTTP/1.1
ost: 39.107.33.96:10000
ontent-Length: 21
ccept: */*
origin: http://39.107.33.96:10000
Requested-With: XMLHttpRequest
ser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
chrome/64.0.3282.186 Safari/537.36
ontent-Type: application/x-www-form-urlencoded; charset=UTF-8
eferer: http://39.107.33.96:10000/
ccept-Encoding: gzip, deflate
ccept-Language: zh-CN,zh;q=0.9
ookie: PHPSESSID=oc0e7gc9nhd8mmgm3cprjc4bp4; td_cookie=18446744072762622588
onnection: close

aram1[]=1&param2[]=2
```

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Wed, 28 Mar 2018 02:20:23 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.20
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Length: 61
Connection: close
Content-Type: text/html

success<script>alert('success!');location.href='/';</script>
```

还没拿到flag。。。。

Md5 Revenge Now!

param1

param2

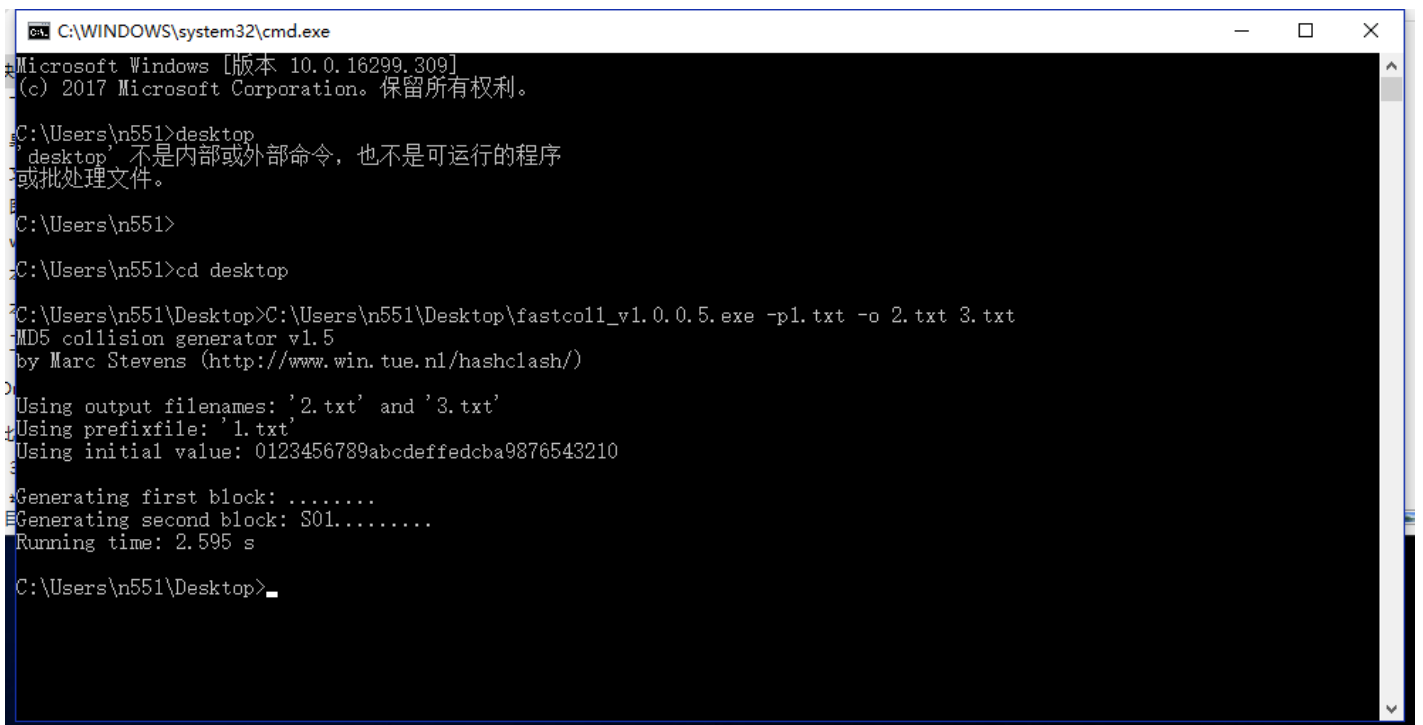
Submit

```
<h2>Md5 Revenge Now!</h2>
<!--
if((string)$_POST['param1']!=(string)$_POST['param2'] && md5($_POST['param1'])==md5($_POST['param2'])){
    die("success!");
}
-->
/-----"-----"-----\
```

它事先定义了string类型了，之前的方法无法绕过去了。

我们用到了fastcoll_v1.0.0.5 这个软件，用来生成两个有着相同MD5值的文件。

先在桌面上建立一个空的文本文件，取名1.txt



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.16299.309]
(c) 2017 Microsoft Corporation. 保留所有权利。

C:\Users\n551>desktop
'desktop' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

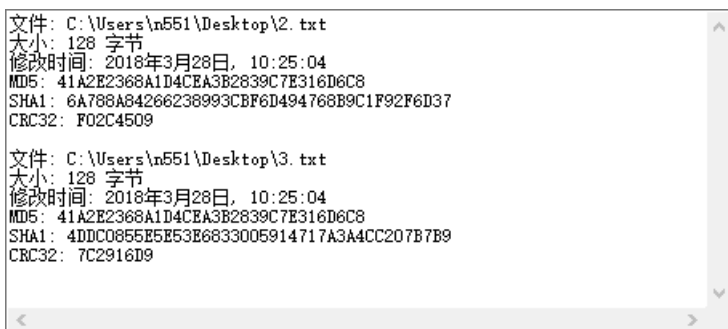
C:\Users\n551>
C:\Users\n551>cd desktop
C:\Users\n551\Desktop>C:\Users\n551\Desktop\fastcoll_v1.0.0.5.exe -p1.txt -o 2.txt 3.txt
MD5 collision generator v1.5
by Marc Stevens (http://www.win.tue.nl/hashclash/)

Using output filenames: '2.txt' and '3.txt'
Using prefixfile: '1.txt'
Using initial value: 0123456789abcdeffedcba9876543210

Generating first block: .....
Generating second block: S01.....
Running time: 2.595 s

C:\Users\n551\Desktop>
```

生成两个文件后，比较它们的md5值



```
文件: C:\Users\n551\Desktop\2.txt
大小: 128 字节
修改时间: 2018年3月28日, 10:25:04
MD5: 41A2E2368A1D4CEA3B2839C7E316D6C8
SHA1: 6A788A84266238993CBF6D494768B9C1F92F6D37
CRC32: F02C4509

文件: C:\Users\n551\Desktop\3.txt
大小: 128 字节
修改时间: 2018年3月28日, 10:25:04
MD5: 41A2E2368A1D4CEA3B2839C7E316D6C8
SHA1: 4DDC0655E5E53E6633005914717A3A4CC207B7B9
CRC32: 7C2916D9
```

发现一模一样

那我们怎么将文件传上去呢？这里将文件转为url编码


```
1 import hashlib
2 import urllib.parse
3
4 print(urllib.parse.quote(open('2.txt','rb').read()))
5 print(urllib.parse.quote(open('3.txt','rb').read()))
```

%C0%B4%FF%1F%BA%00%A6%C4%E0%1D%BF%5B%29%87%F5%G%BDZzNnD%3D%24%17%9E%82%CD%CD%C9%AD%0FZ%09%02%C0%DDi%23%B6%EB%0714%EF%CF%95%5B%29%A0I%C9Rq%7D1%C6%23N%7Dk%21Ei%92%F7%FA%04%AA%80%24%8C%B4D%CBxEB%D6%E5%0C%EF%1D%A6%01%A6Ff%D1Z%867%EC%C7%FDcpE%1E%14%F4%0F%04%84%98B%3C%D7%A6Dv%AEIa%25%DA%ED%3B%16E%10%A1%C8/%92%7D%F9%5C

[Finished in 0.5s]

再用burpsuite抓一下包，修改一下参数

The screenshot shows the Burp Suite interface with the 'Request' tab selected. The request is a POST to /HTTP/1.1 with a host of 39.107.33.96:10000. The request body contains a long URL-encoded parameter. The 'Response' tab shows a 200 OK status and a message: 'success! flag is QWB{s1gns1gns1gnaftermd5}'.

flag is QWB{s1gns1gns1gnaftermd5}

转载于:<https://www.cnblogs.com/P201521410043/p/8662354.html>