

CTF——Web——MD5漏洞

原创

[Captain Hammer](#) 于 2019-08-09 16:33:08 发布 1306 收藏 9

分类专栏: [web安全 CTF 类型题总结](#)

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/vhkjhws/article/details/97618629>

版权



[web安全](#) 同时被 2 个专栏收录

19 篇文章 5 订阅

订阅专栏



[CTF 类型题总结](#)

11 篇文章 35 订阅

订阅专栏

md5 (\$ss, ture/false) 漏洞:

ture : 为16位, false: 为32位

第一种

```
$_GET['a'] != $_GET['b']
&&
MD5($_GET['a']) == MD5($_GET['b'])
```

要让上面的等式成立, a和b的值不能相等, 但是md5后的值相等。(php弱类型) 因为是 == 比较, 只判断值是否相等, 不判断类型是否相同。如果类型不同先转换为相同类型再进行比较而PHP在处理哈希字符串时, 会把0E开头的哈希值解释为0。所以如果两个值通过md5后值都已0E开头, 就会相等。

那么这些值有哪些呢?

```
QNKCDZO
0e830400451993494058024219903391

s878926199a
0e545993274517709034328855841020

s155964671a
0e342768416822451524974117254469

s214587387a
0e848240448830537924465865611904

s214587387a
0e848240448830537924465865611904

s878926199a
0e545993274517709034328855841020
```

第二种

```
$_POST['a1']!= $_POST['a2']
&&
md5($_POST['a1'])===md5($_POST['a2'])
```

=== 不仅比较值相等还会要求类型比较

但是

md5无法处理数组！所以构建数组就可以了

第三种

```
(string)$_POST['a1']!=(string)$_POST['a2']
&&
md5($_POST['a1'])===md5($_POST['a2'])
}
```

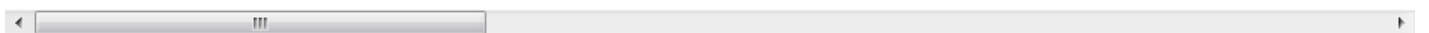
这里比较的是字符串

那么md5值相同的字符串有哪些呢？

#仔细看，数值均不同

#强网杯某大牛wp

```
$Param1="\x4d\xc9\x68\xff\x0e\xe3\x5c\x20\x95\x72\xd4\x77\x7b\x72\x15\x87\xd3\x6f\xa7\xb2\x1b\xdc"
$Param2="\x4d\xc9\x68\xff\x0e\xe3\x5c\x20\x95\x72\xd4\x77\x7b\x72\x15\x87\xd3\x6f\xa7\xb2\x1b\xdc"
#008ee33a9d58b51cfb425b0959121c9
```

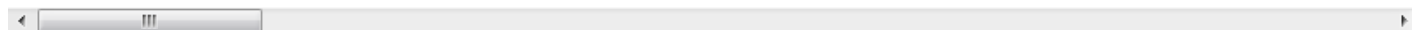


#知乎 Believe

```
$data1="\xd1\x31\xdd\x02\xc5\xe6\xee\xc4\x69\x3d\x9a\x06\x98\xaf\xf9\x5c\x2f\xca\xb5\x07\x12\x46\x
```

```
$data2="\xd1\x31\xdd\x02\xc5\xe6\xee\xc4\x69\x3d\x9a\x06\x98\xaf\xf9\x5c\x2f\xca\xb5\x87\x12\x46\x
```

```
#79054025255fb1a26e4bc422aef54eb4
```



新的加密函数

sha1 () 漏洞:

sha1 () 也不能处理数组, 处理数组会返回false,