CatfishCMS-4-4.6.15-xss到getshell-包含漏洞的利用

**0x00 前言**

因为这个cms更新了版本导致之前的漏洞无法使用了，更新一波其他的利用方式。漏洞只做审计思路交流，请勿用于实战。

这个链接有源码与过程与在测试漏洞中使用到的工具等等

链接：https://pan.baidu.com/s/1jHC2igu 密码：7qa7

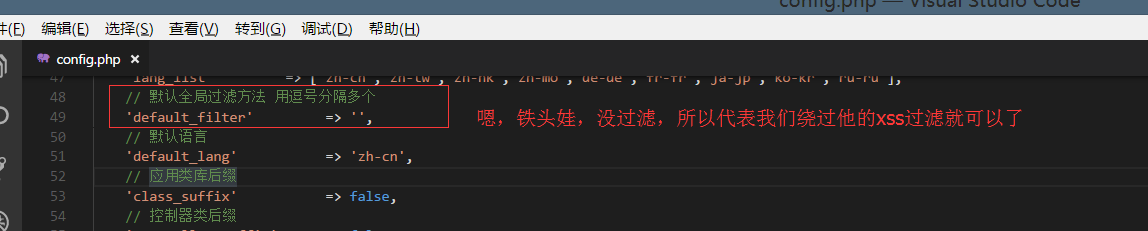
**0x01 介绍**

我们还是一样的，先查看他是否有全局过滤。

文件：application\config.php

参数：default\_filter

TP是可以在config这里设置全局过滤的，所以我们一定要先看是否拥有全局过滤

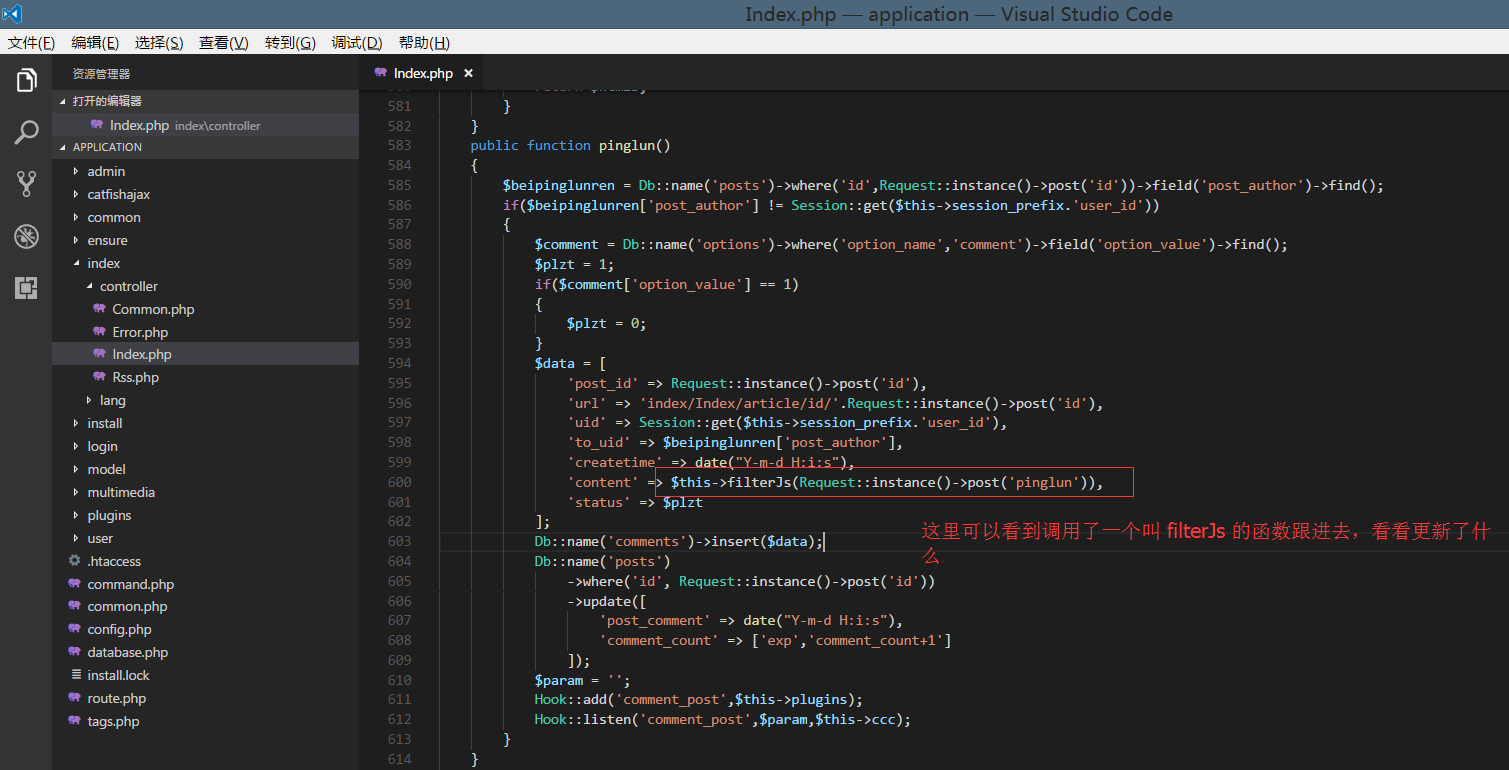


**0x02 漏洞1：又是-用户评论处xss**

url：http://127.0.0.1/cms/CatfishCMS-4.6.12/index.php/index/Index/pinglun

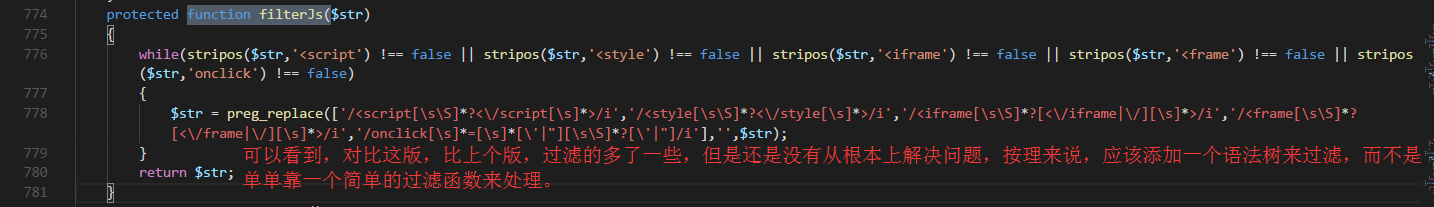
文件：application/index/controller/Index.php

方法：pinglun()



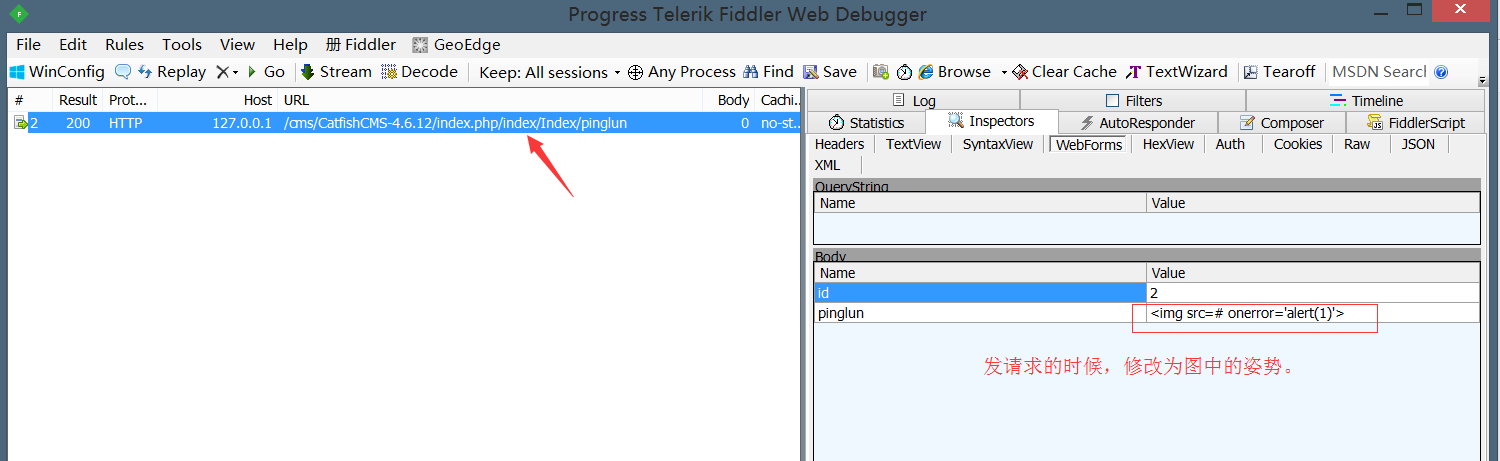
文件：application\index\controller\Common.php

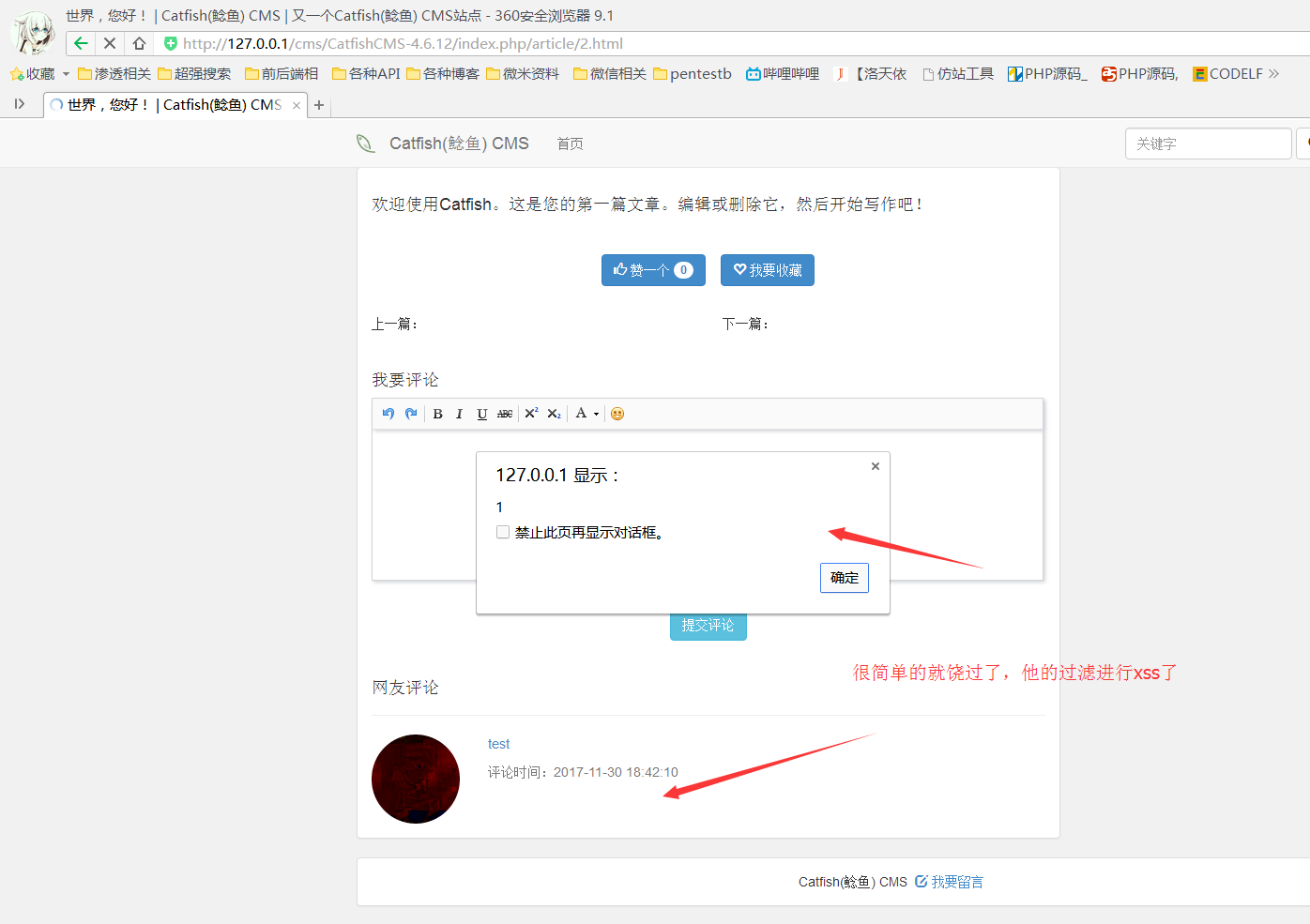
过滤函数：filterJs()

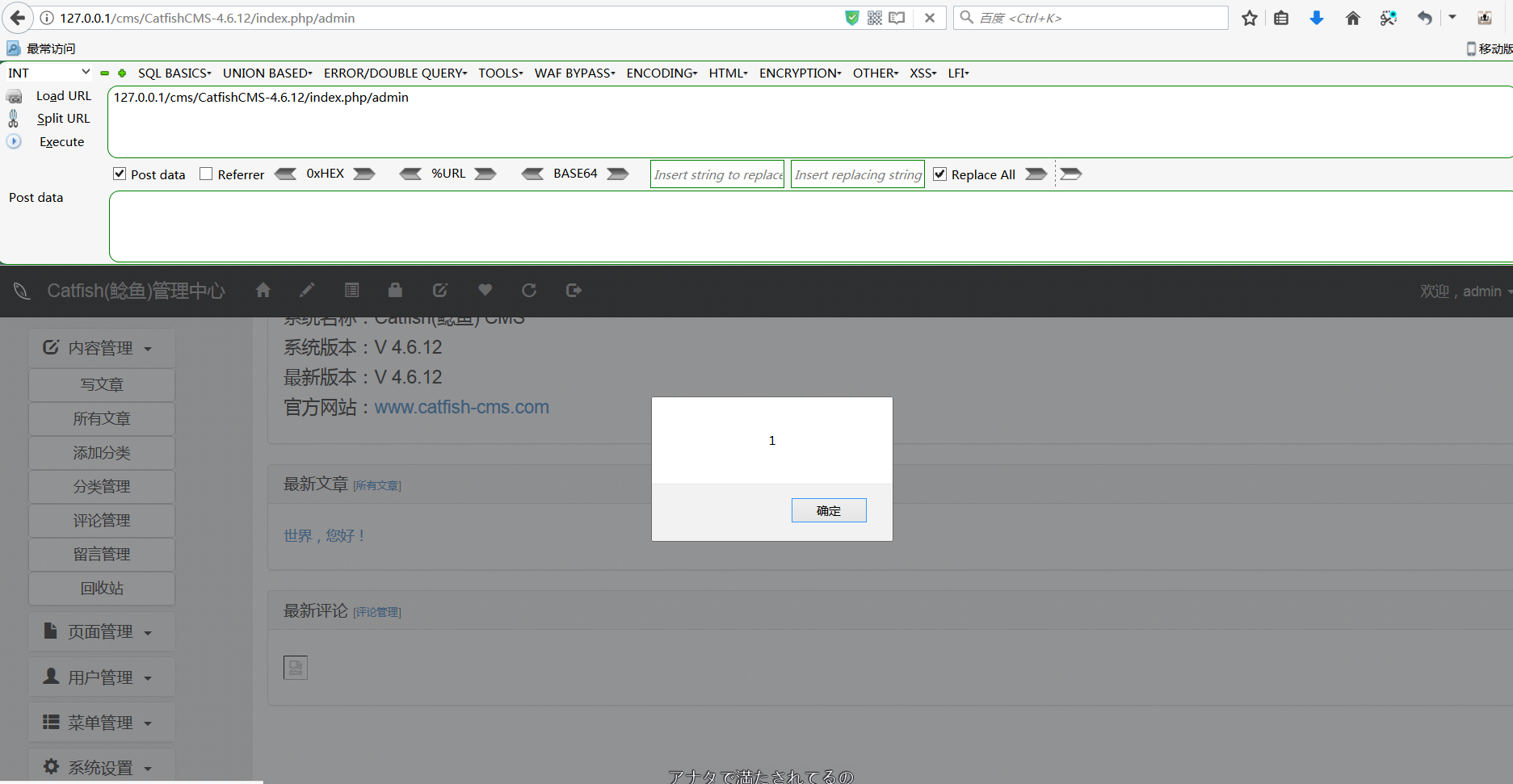


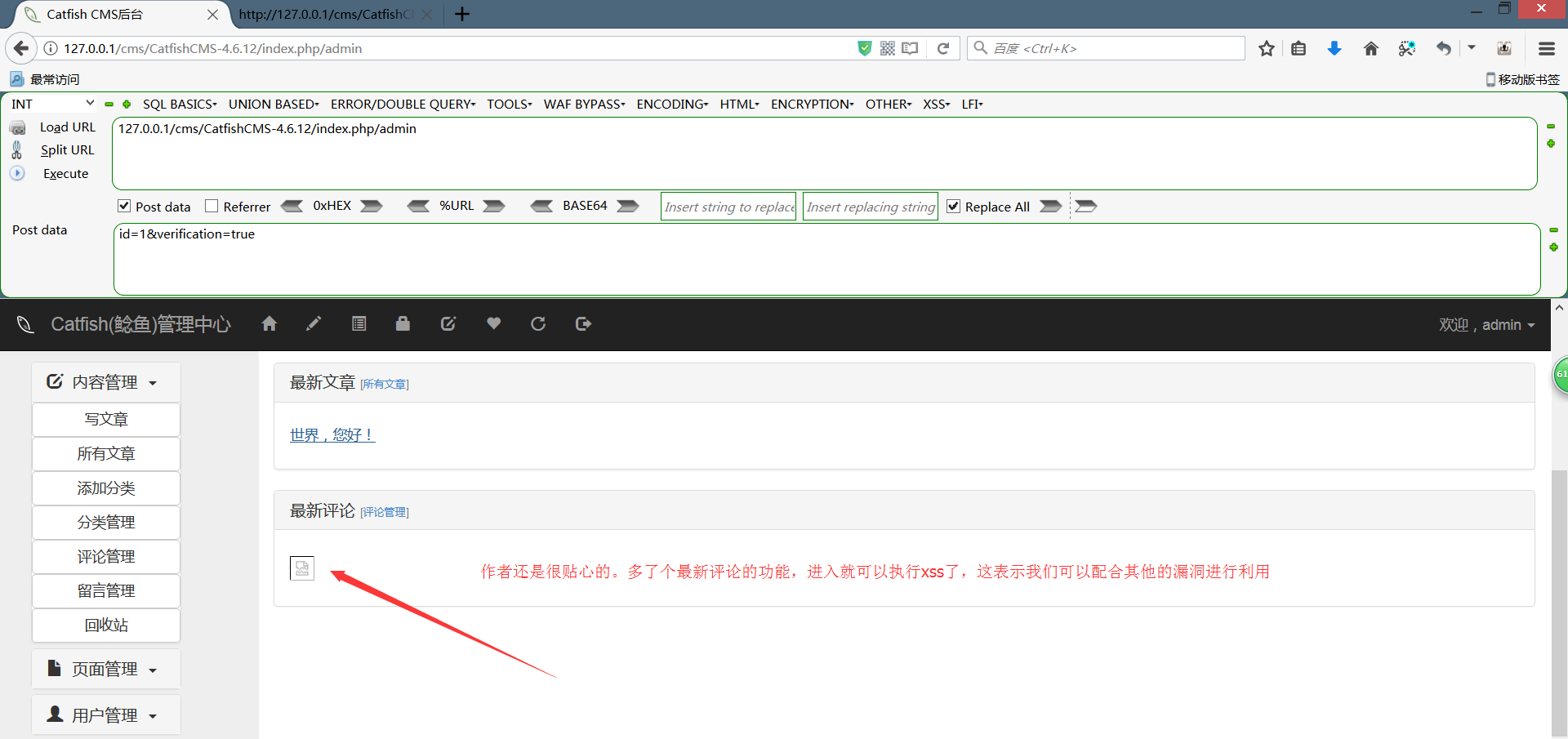
还是一样的注册个账户先。











**0x03 图片马的生成：**

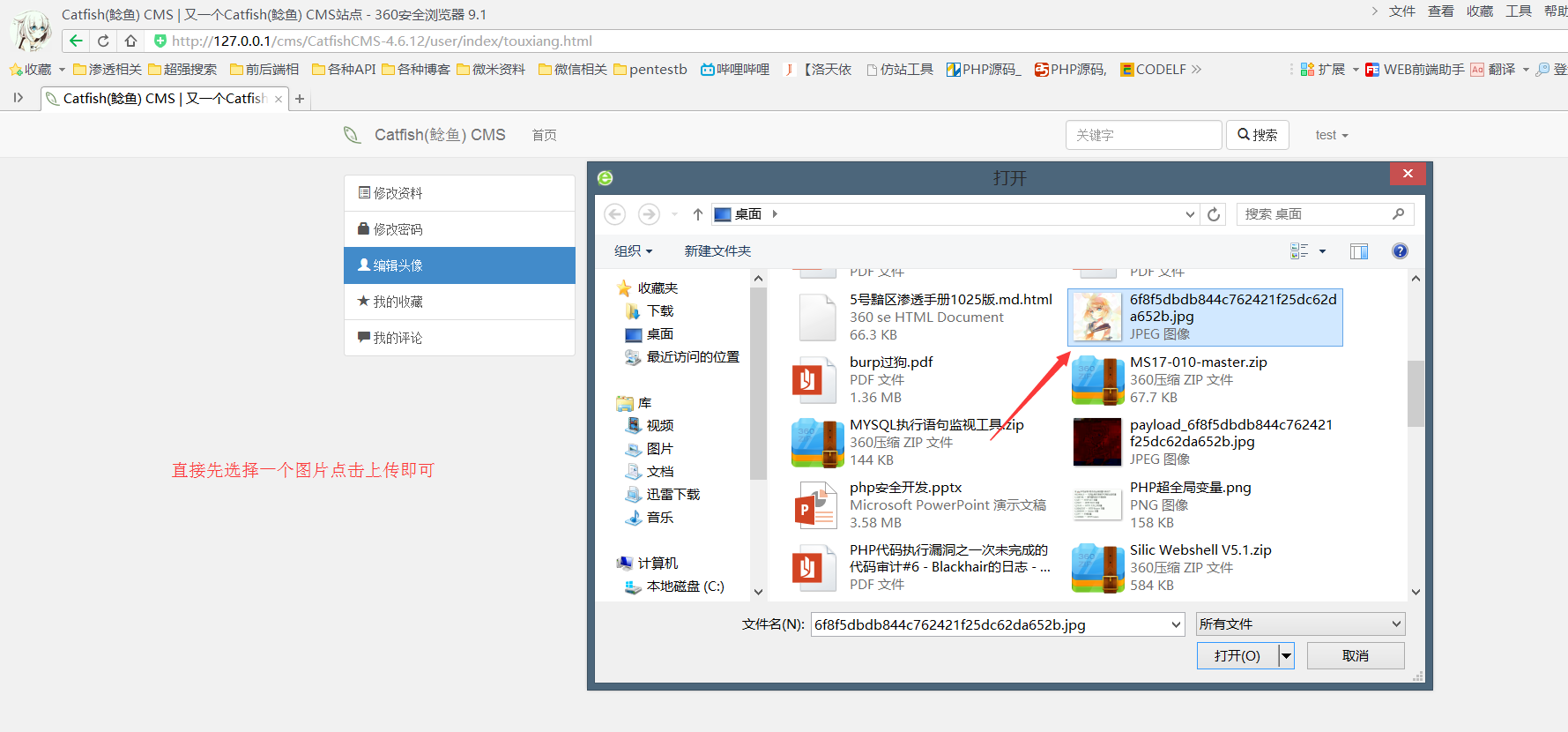
利用这个漏洞之前我们先添加一个图片马。

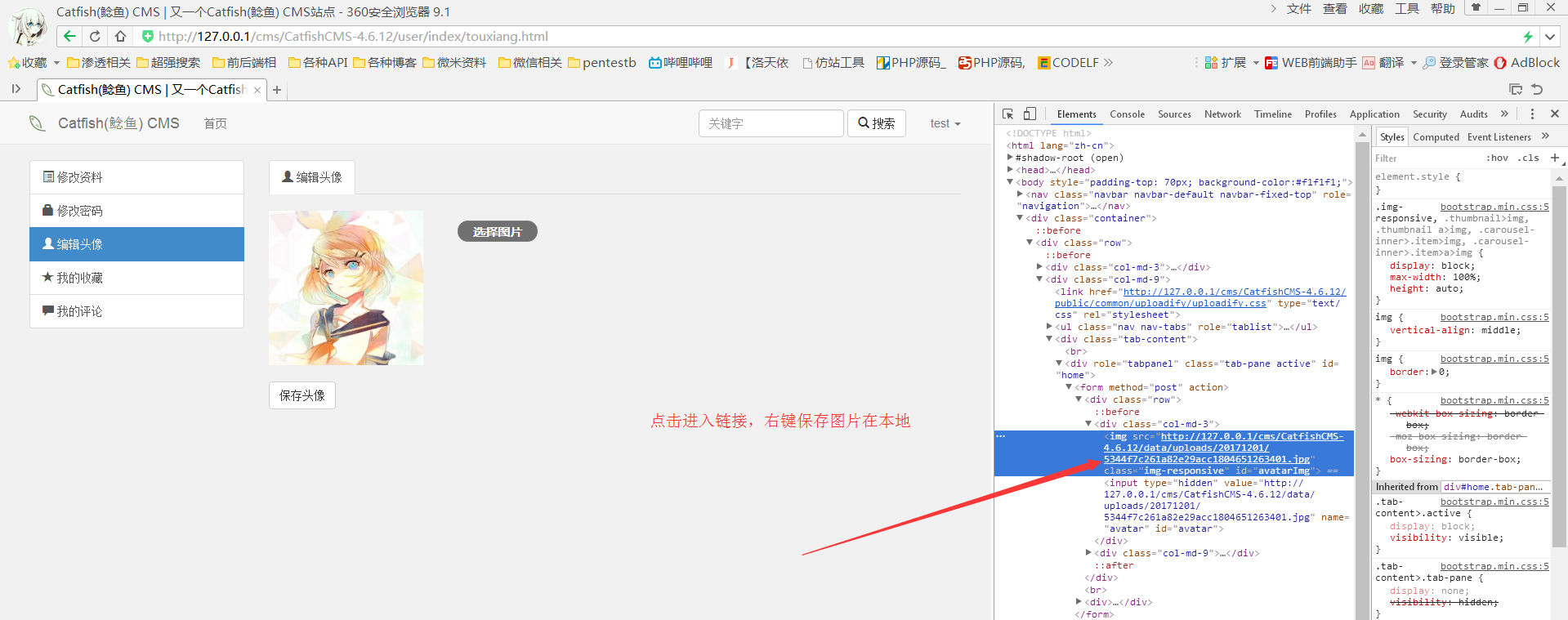
图片马生成的代码会一起放在后面的压缩包中。

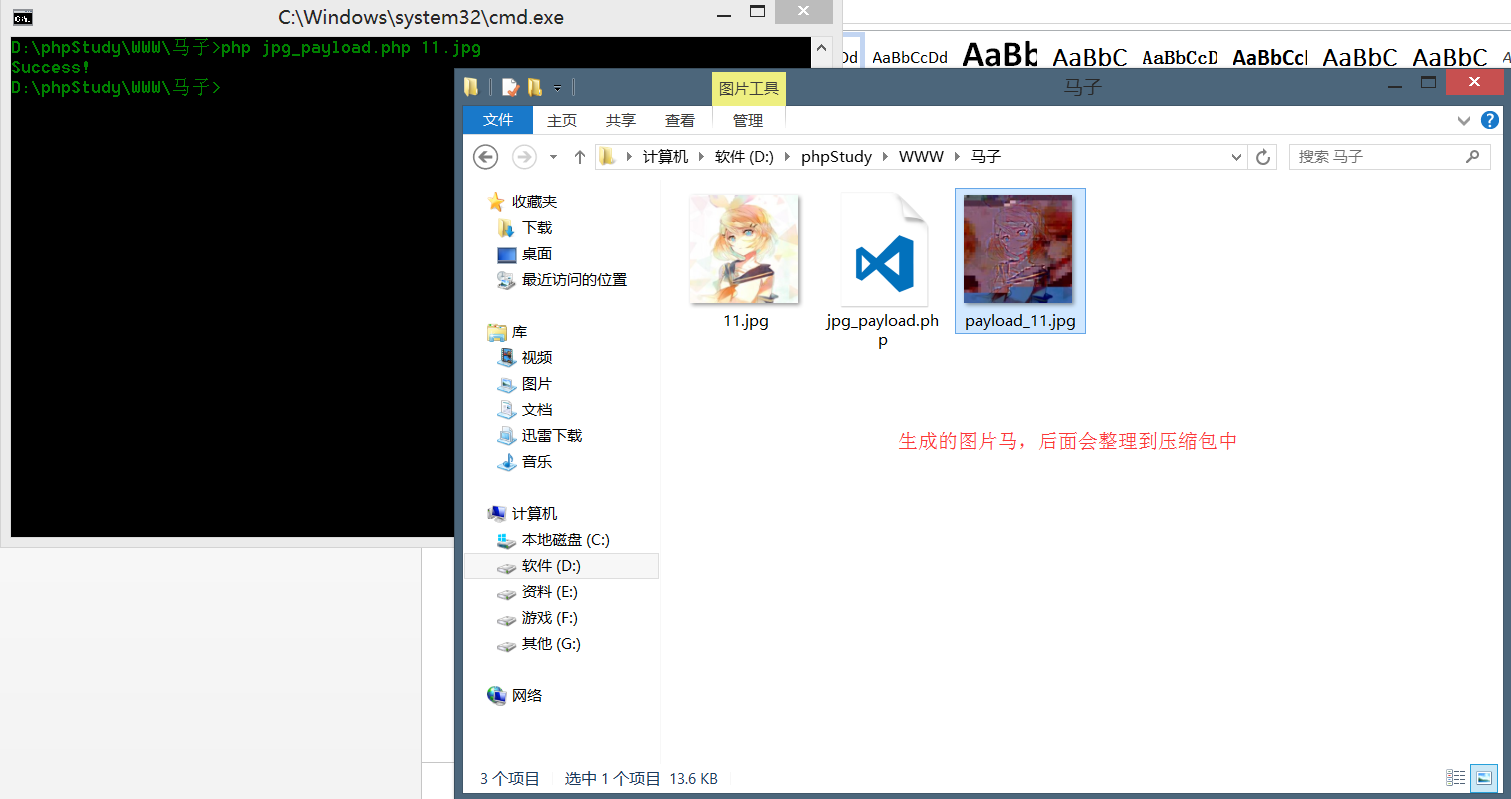
首先说说为什么要添加图片马因为要利用这个文件包含，我们必须要写入个可以包含的恶意文件先。那么图片是每个网站比较通用的东西。那么在图片里面写马子无疑是一个比较通用的方法。

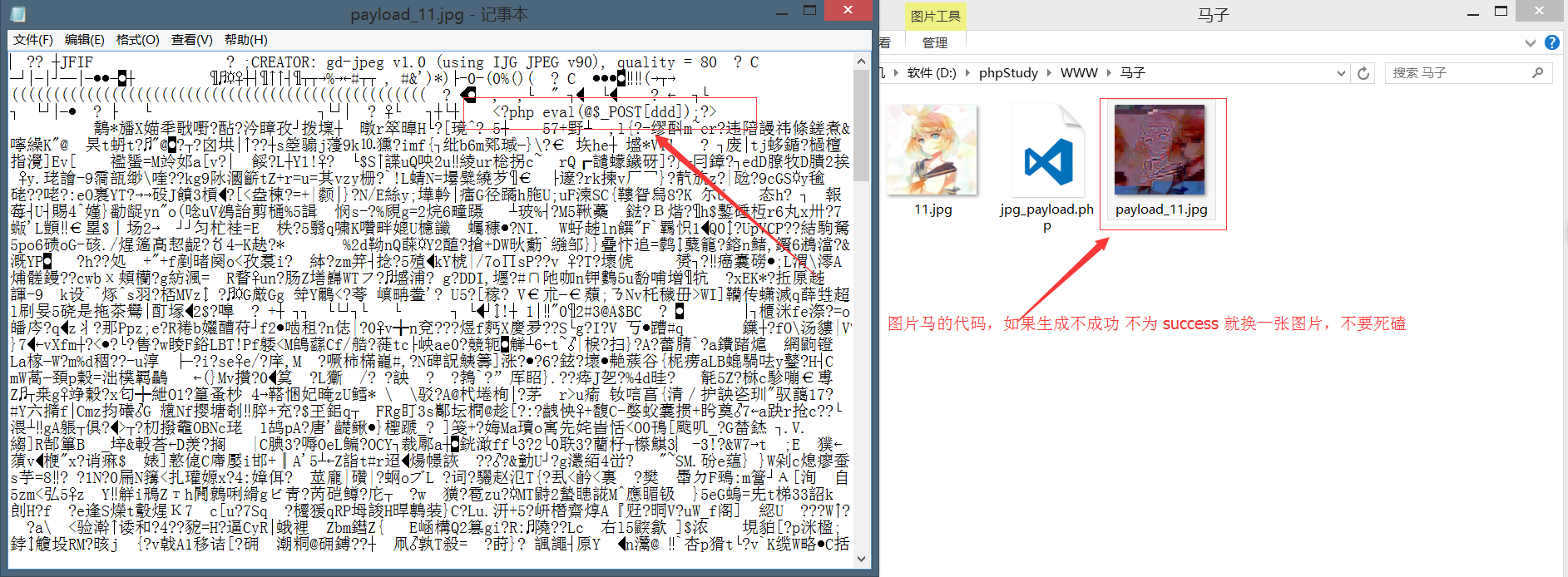
首先感谢先知安全的文章，没有先知的文章还真的绕不过去。。。  
这里确实不要多说什么，因为文件上传以后 php-GD对图片的渲染和处理会导致webshell代码错位失效，所以我们需要进行绕过。  
绕过php-GD对图片的渲染和处理导致webshell代码错位失效(此处参考索马里海盗方法)  
图片会经过php-GD处理，会导致webshell语句错位失效，如何在处理后仍然保留shell语句呢？  
在正常图片中插入shell并无视GD图像库的处理，常规方法有两种  
    1.对比两张经过php-gd库转换过的gif图片，如果其中存在相同之处，这就证明这部分图片数据不会经过转换。然后我可以注入代码到这部分图片文件中，最终实现远程代码执行  
    2.利用php-gd算法上的问题进行绕过  
这里我们选择第二种，使用脚本进行处理图片并绕过  
        1、上传一张jpg图片，然后把网站处理完的图片再下回来 比如x.jpg  
        2、执行图片处理脚本脚本进行处理 php jpg\_payload.php x.jpg  
        3、如果没出错的话，新生成的文件再次经过gd库处理后，仍然能保留webshell代码语句  
提示：  
        1、图片找的稍微大一点 成功率更高  
        2、shell语句越短成功率越高  
        3、一张图片不行就换一张 不要死磕  
注：上面的字全部是抄的，先说明一下不然给人按在地上骂就不好了

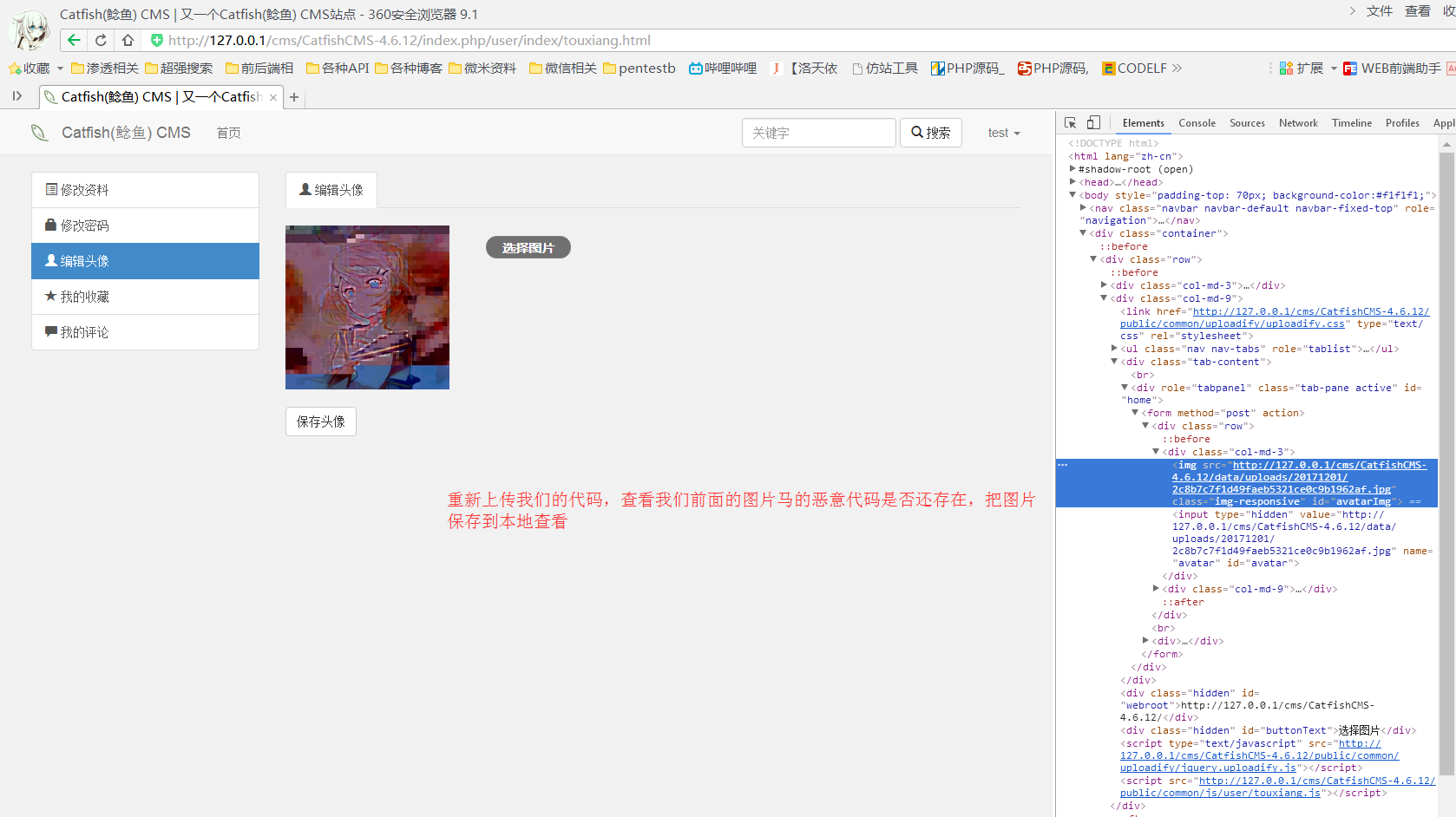
制作过程：

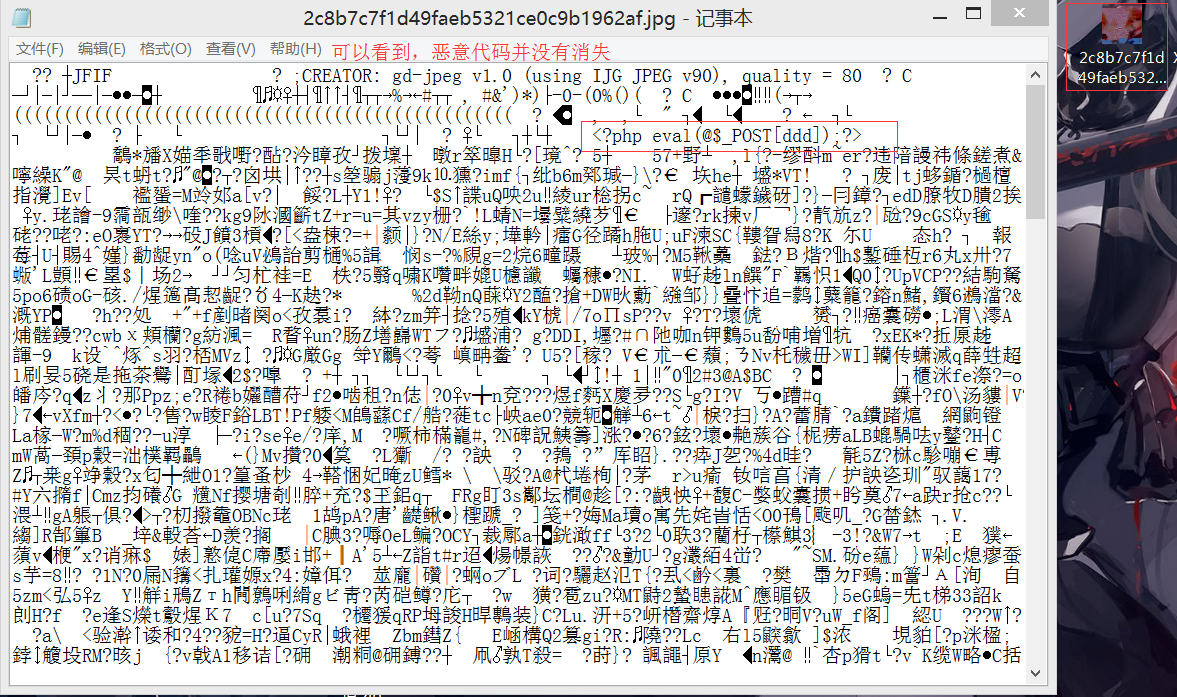






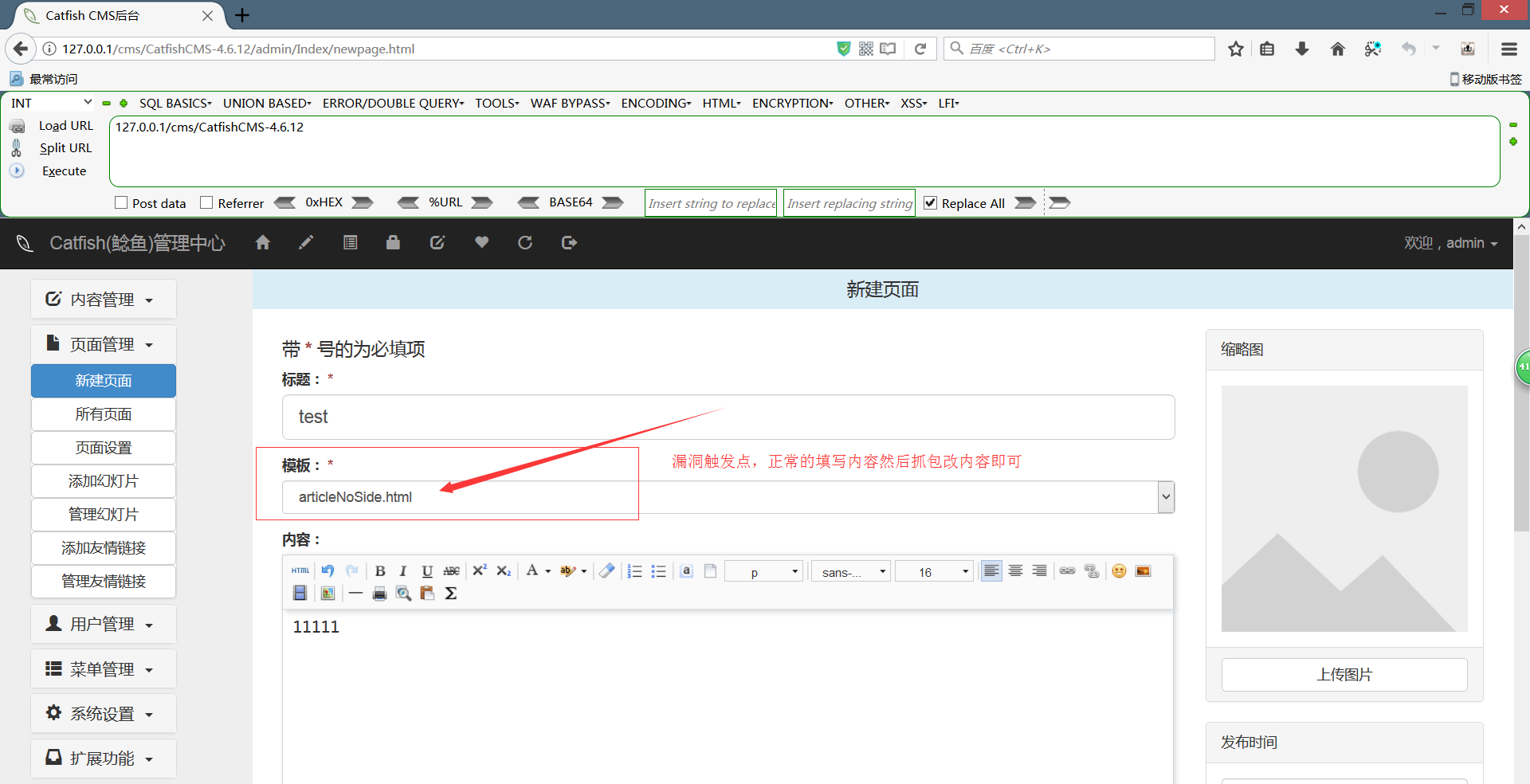


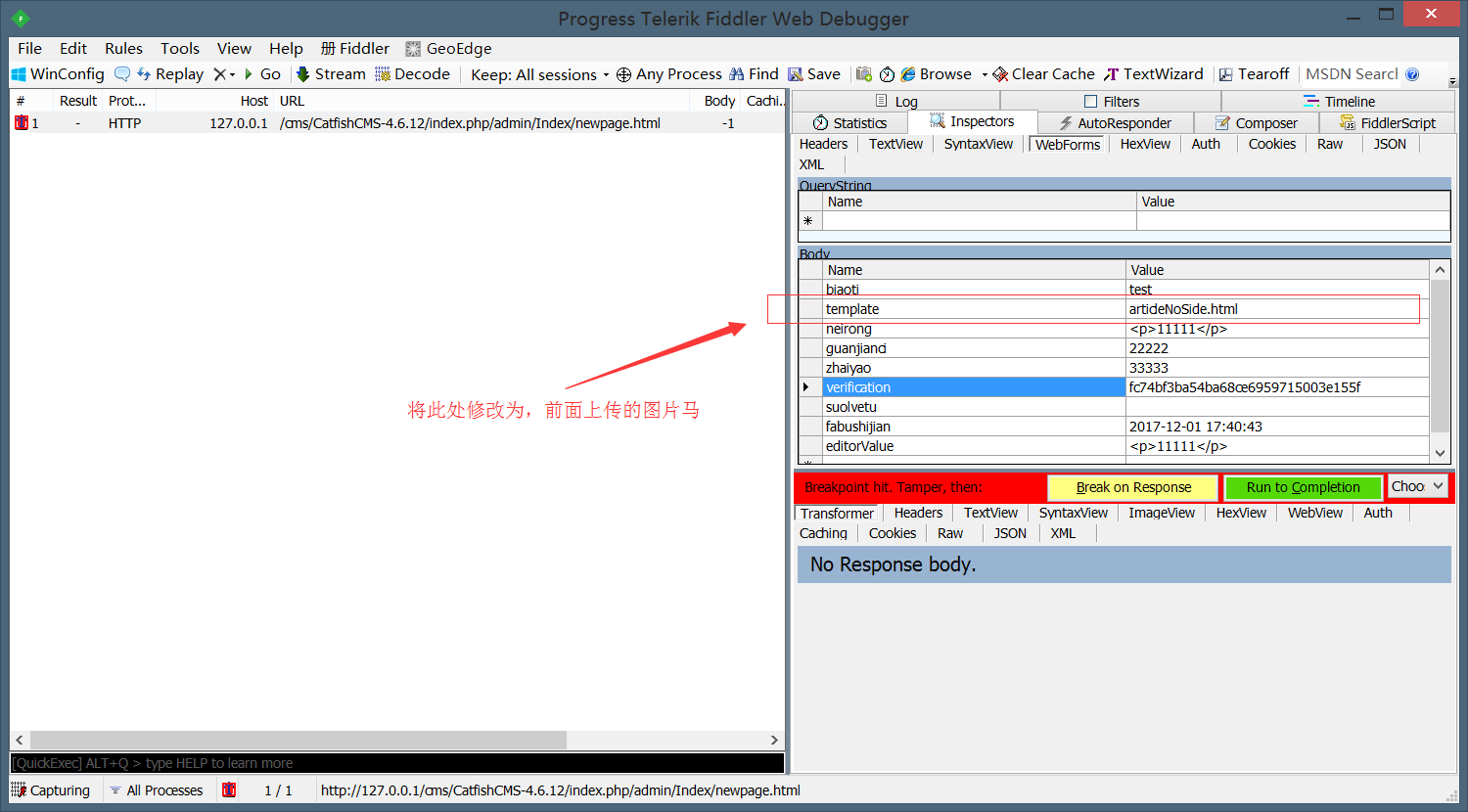


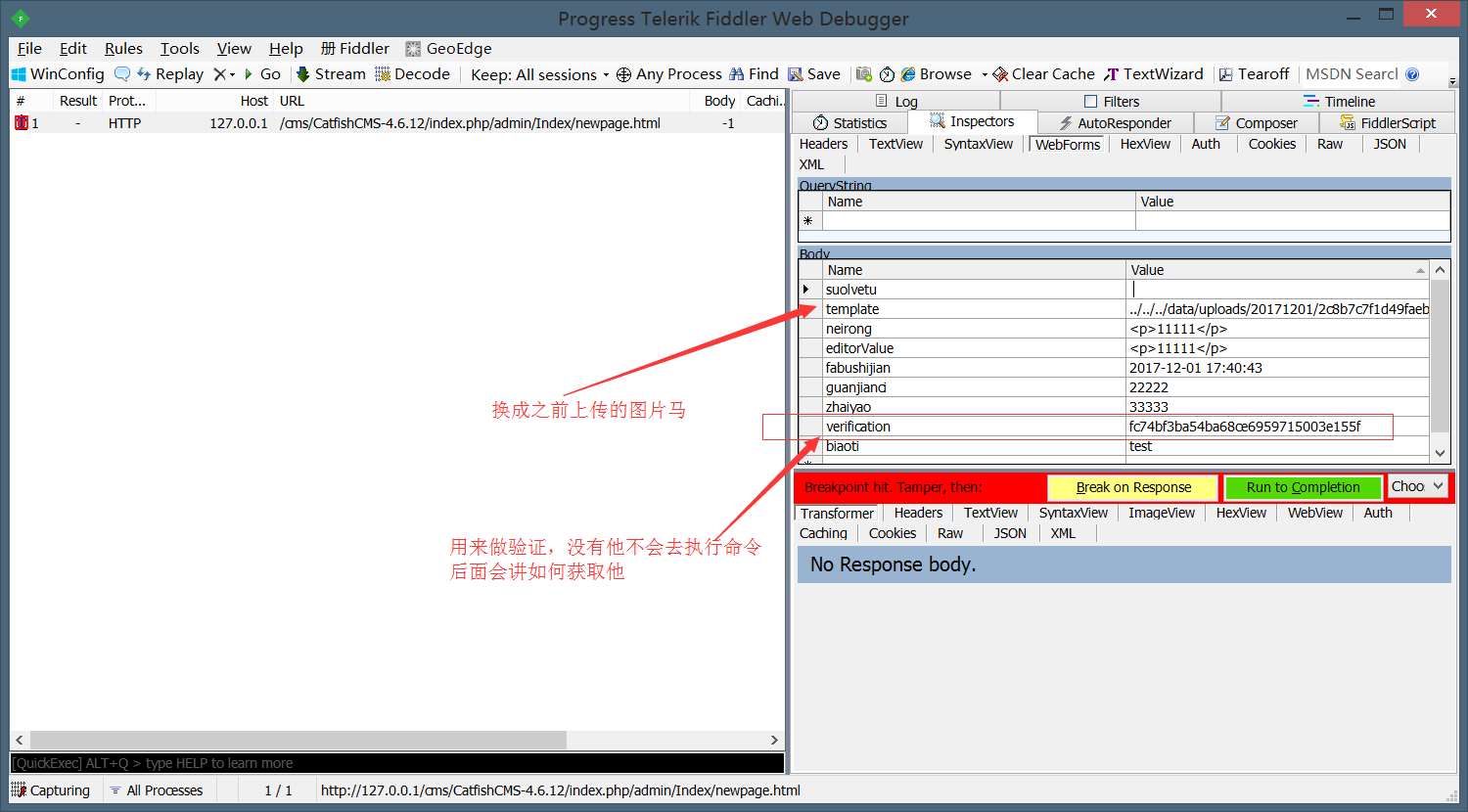


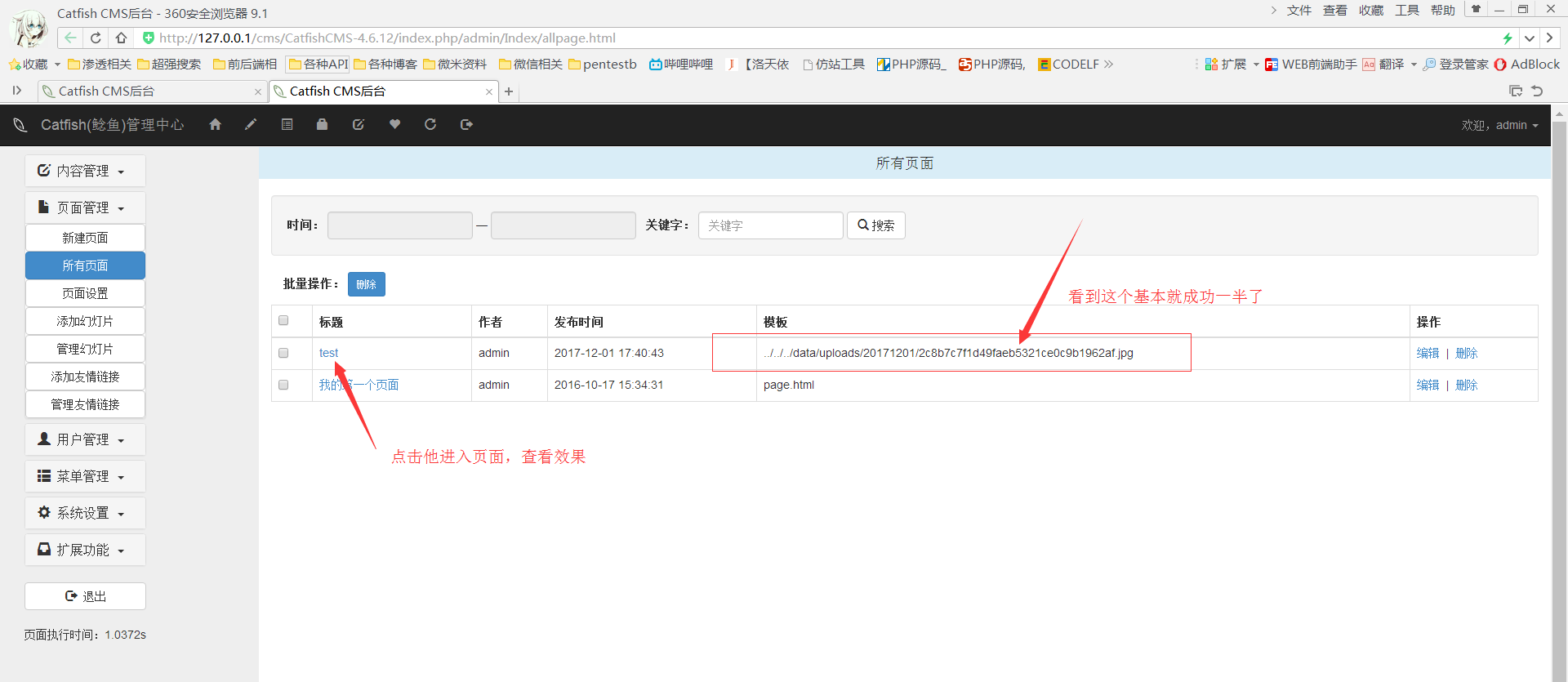
**0x04 使用TP的模版函数进行文件包含(很明显的开发问题)：**

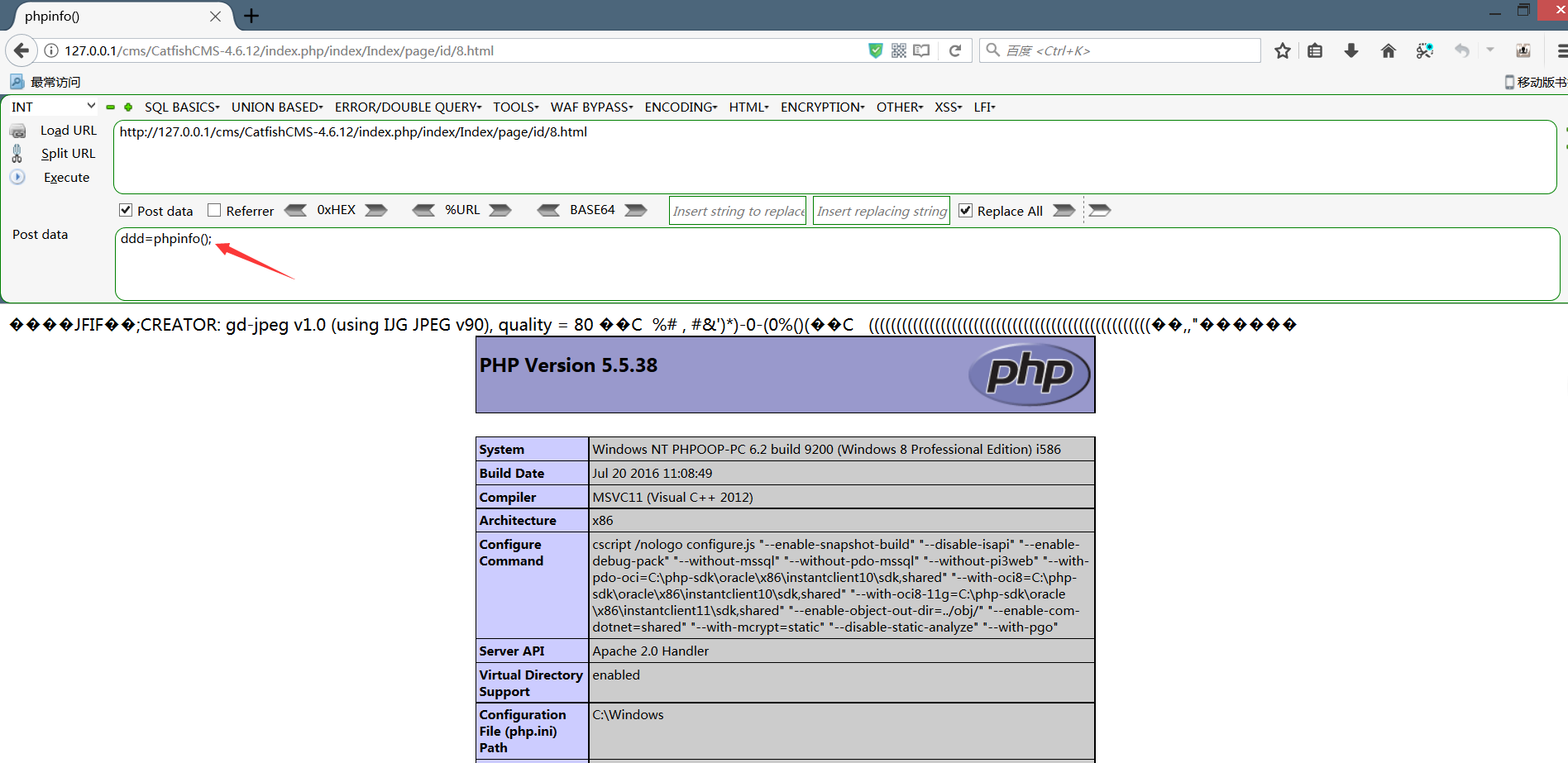
先说明这个锅TP绝对不背，肯定是开发者的问题。









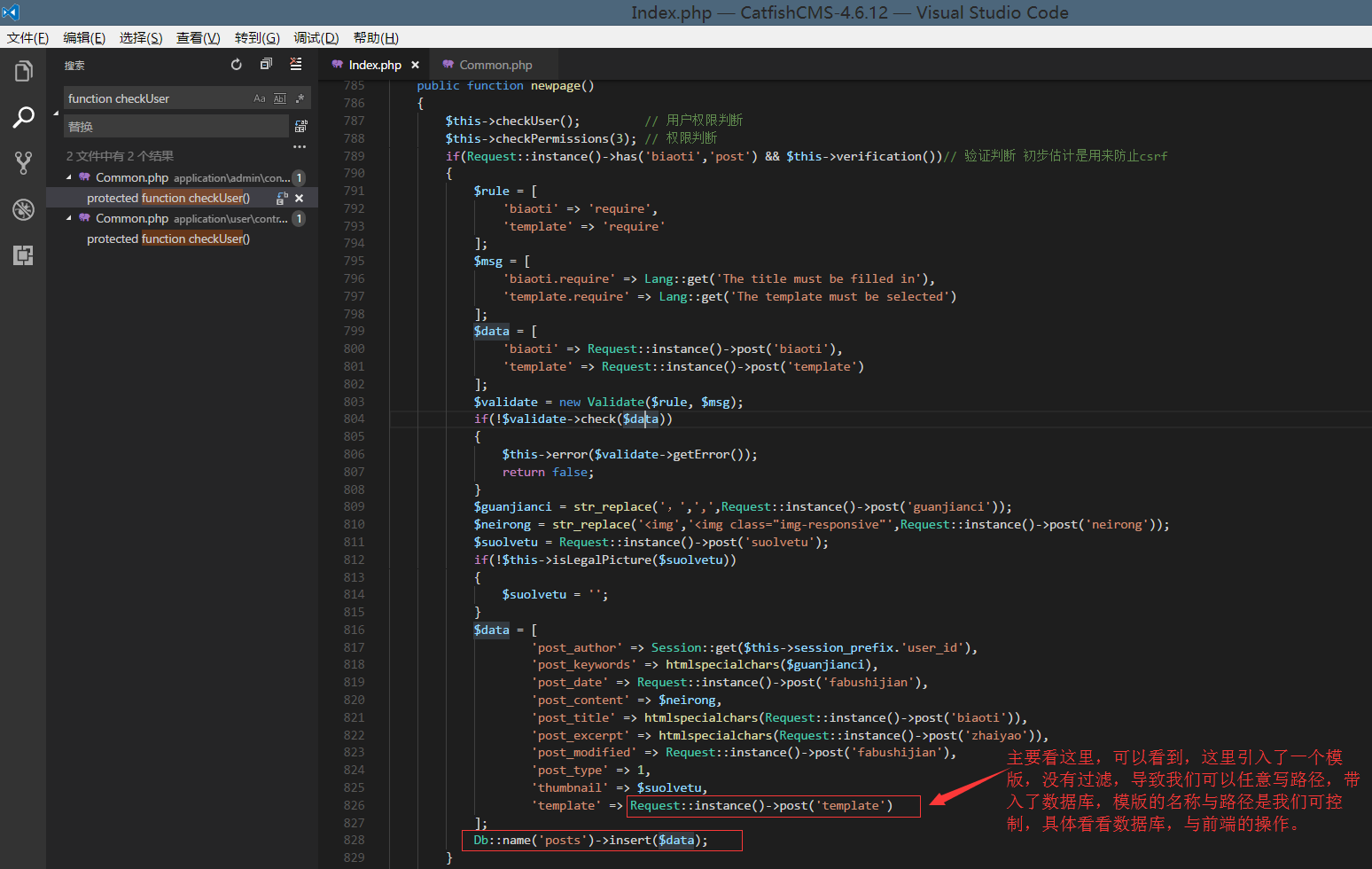


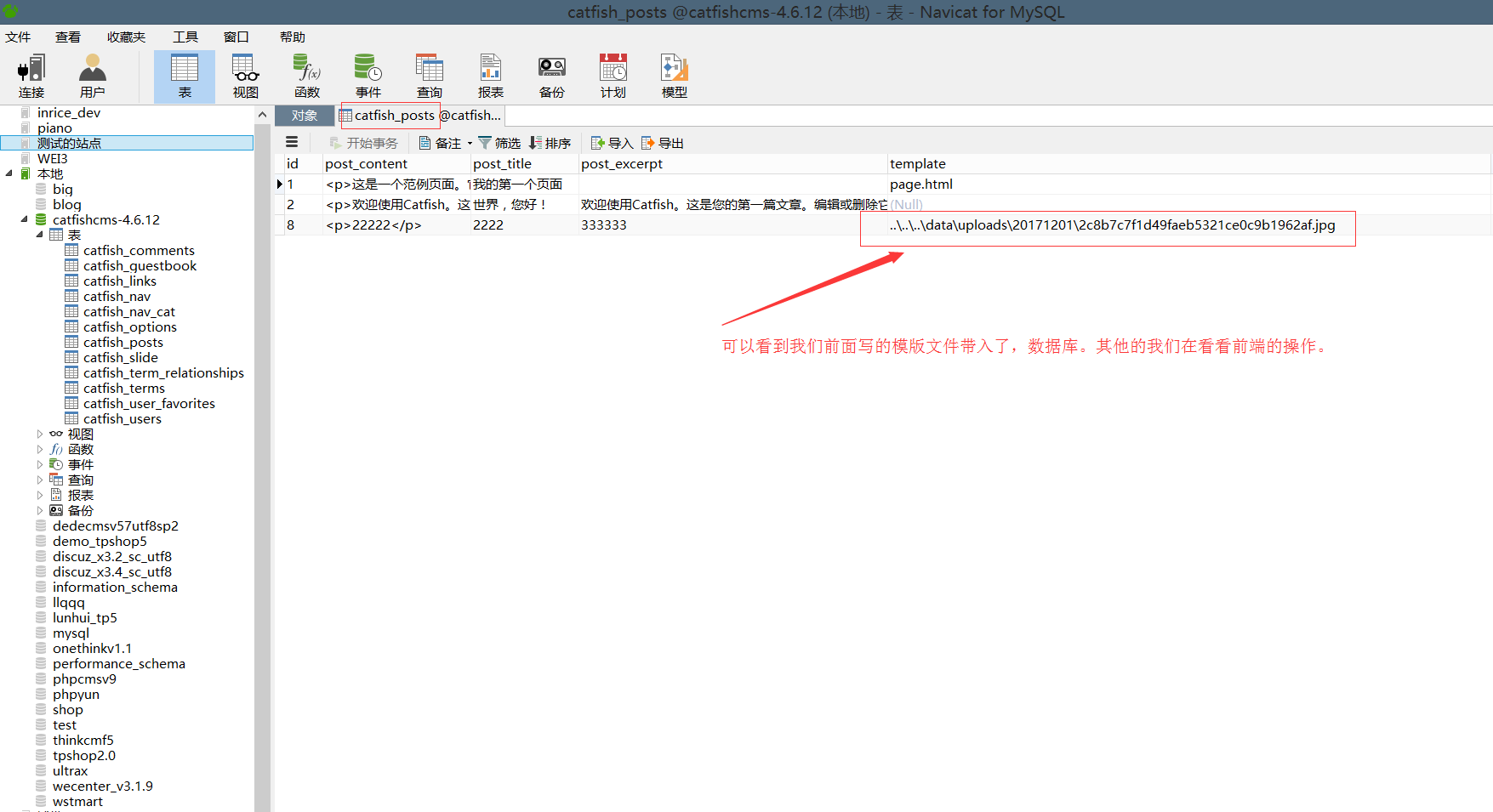
漏洞原理：

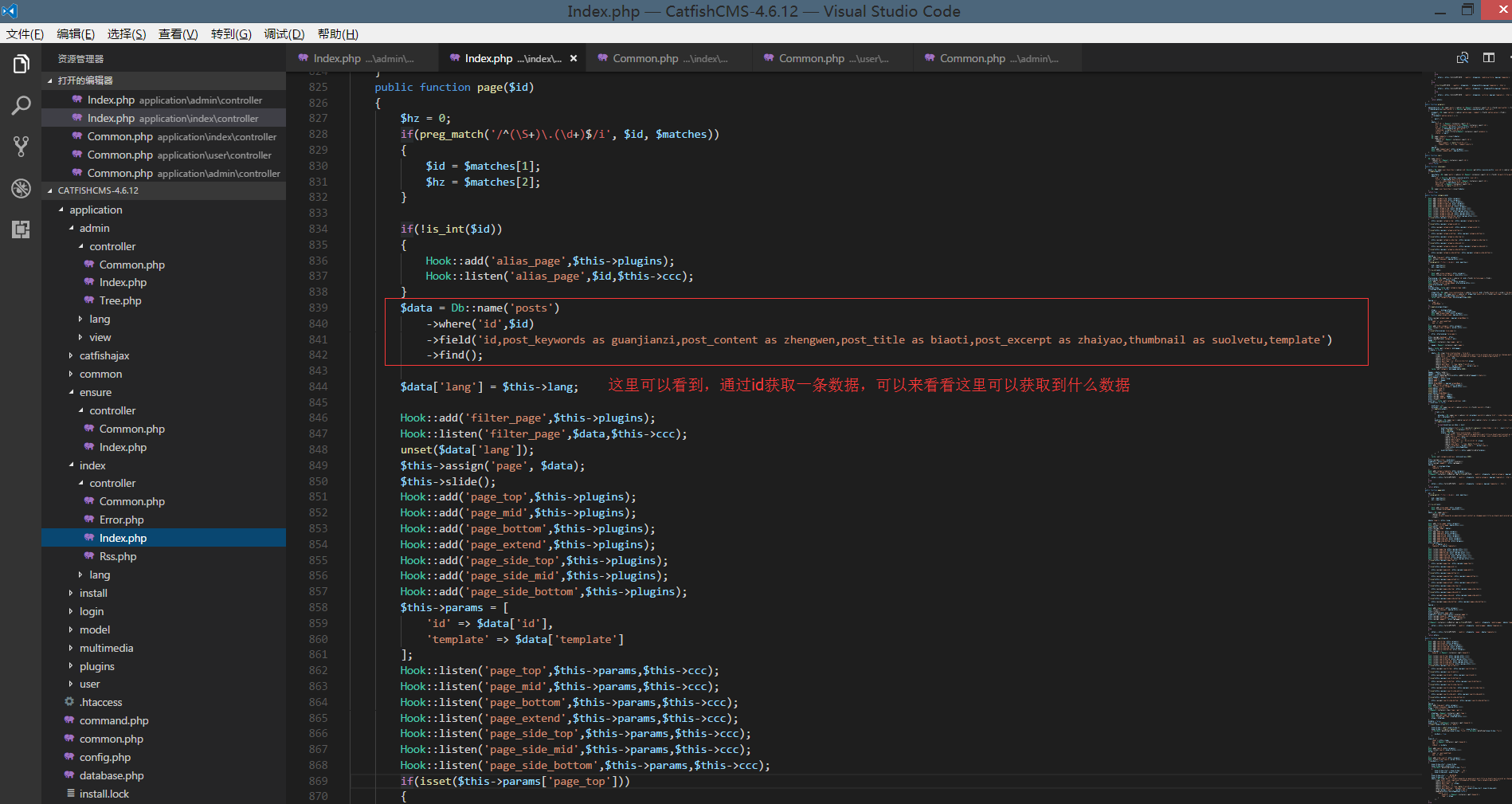
url：http://127.0.0.1/cms/CatfishCMS-4.6.12/index.php/admin/Index/newpage.html

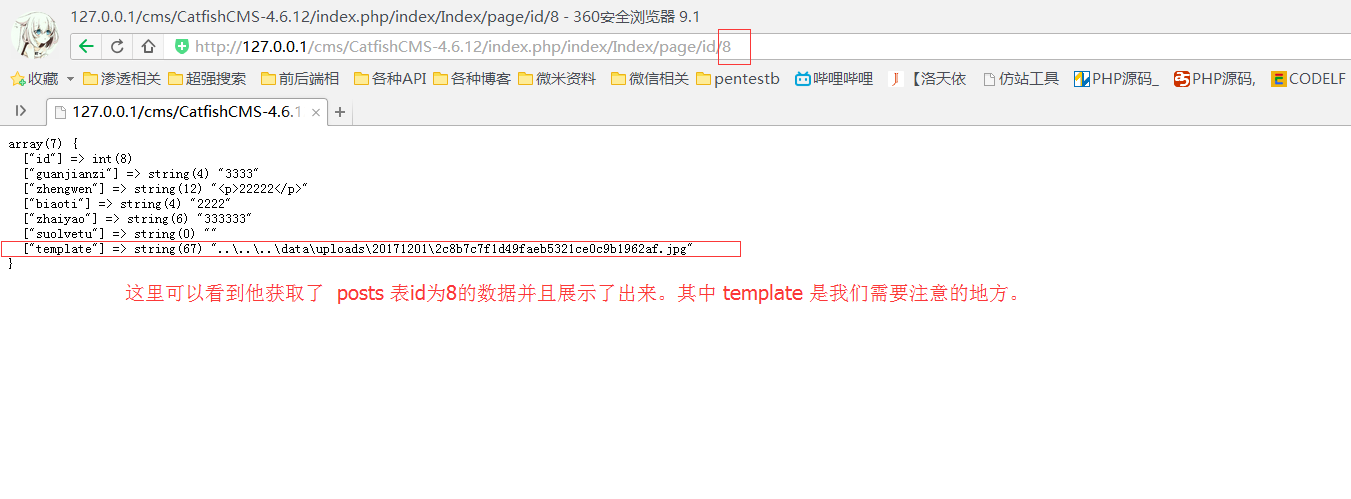
文件地址：\CatfishCMS-4.6.12\application\admin\controller\Index.php

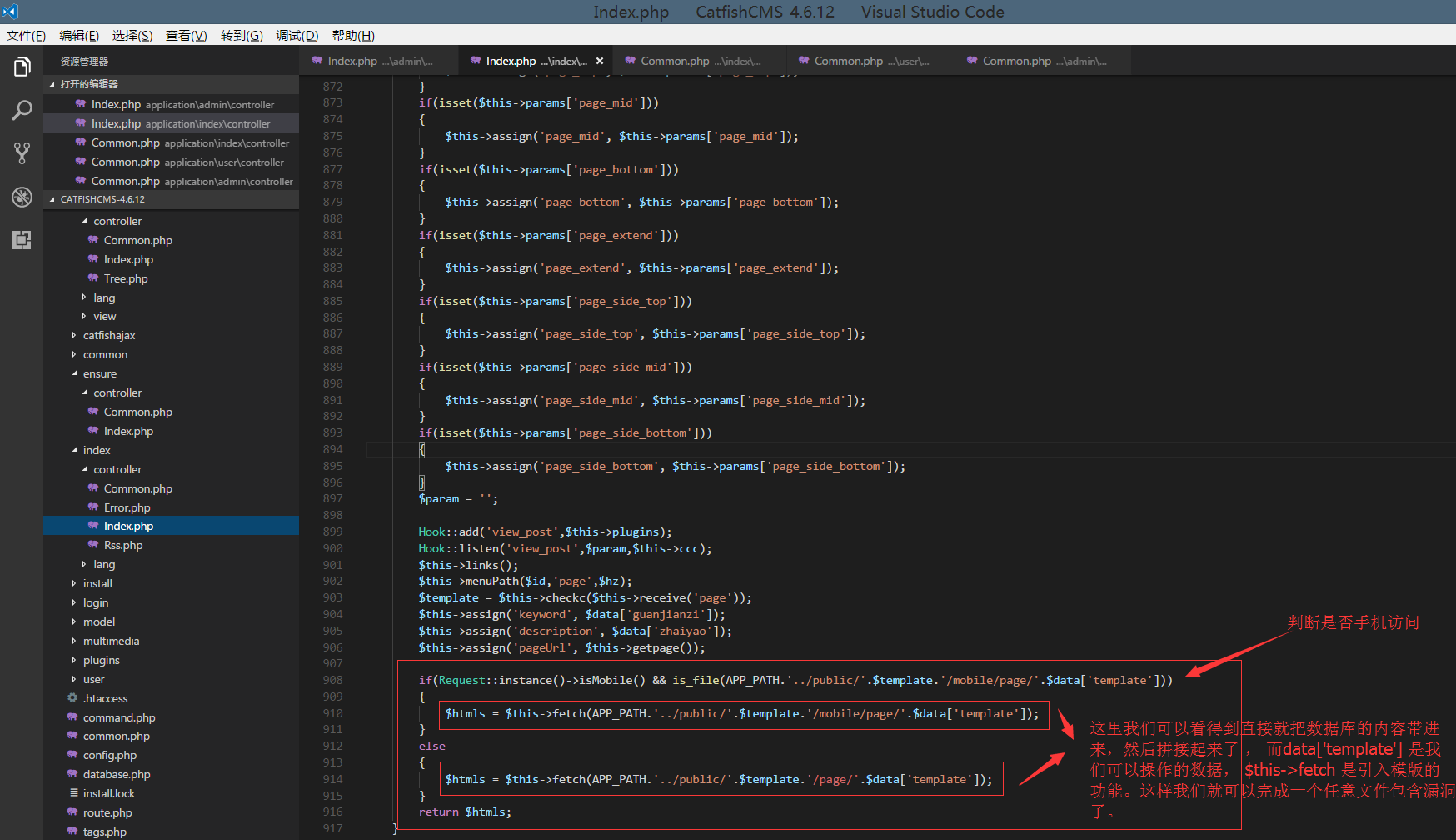
函数：newpage()









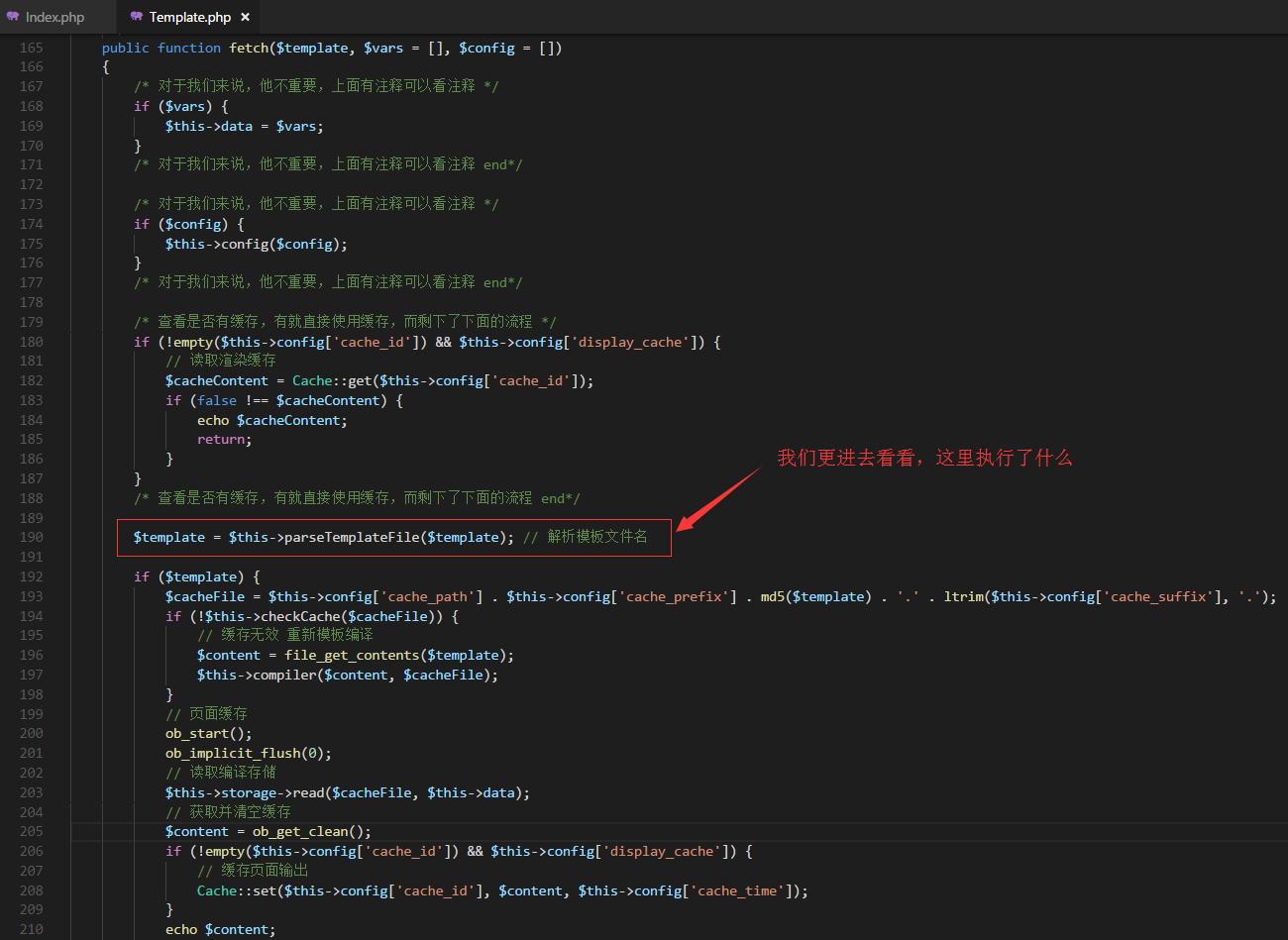


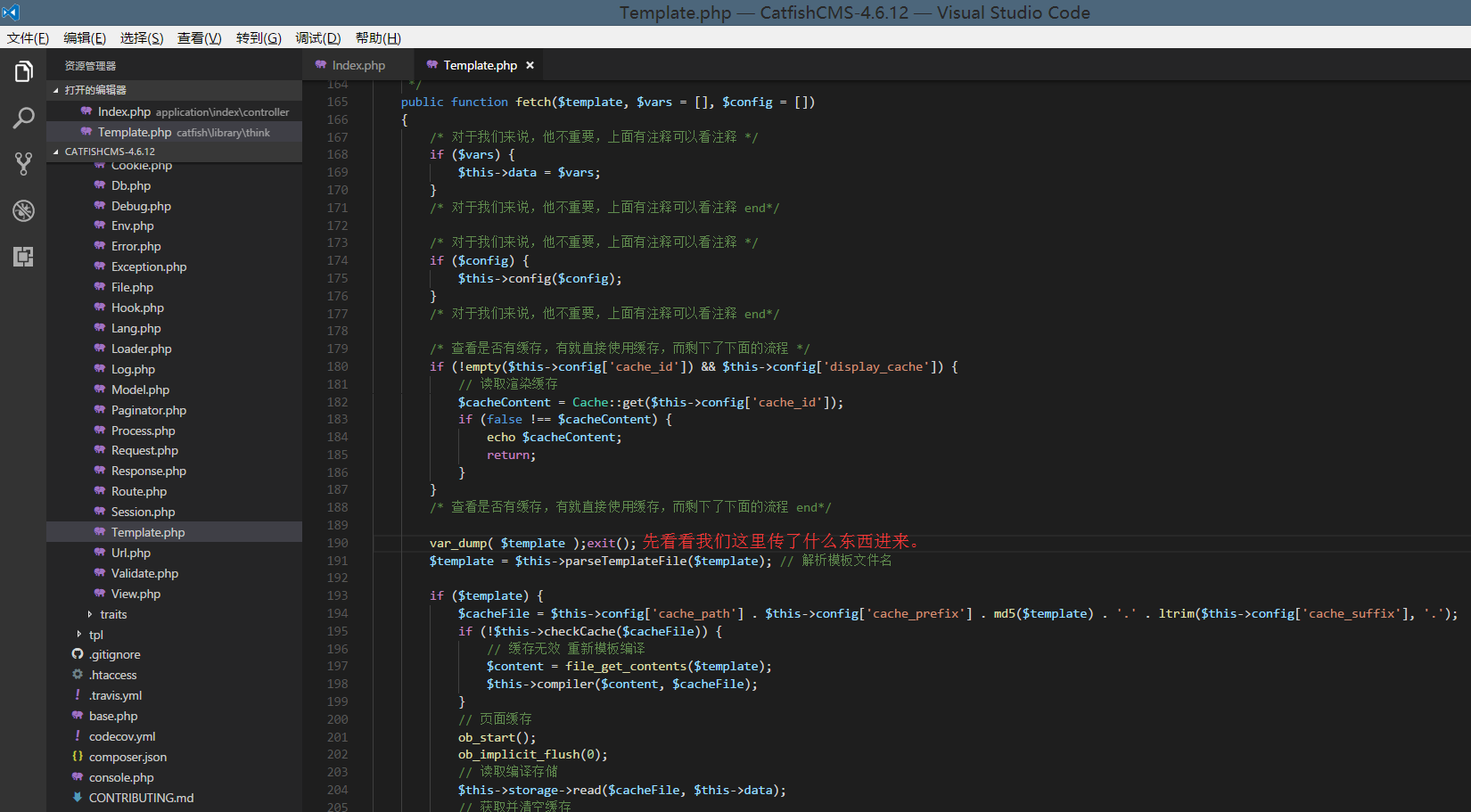
**0x05 TP的模版函数-fetch方法**

文件：CatfishCMS-4.6.12\catfish\library\think\Template.php

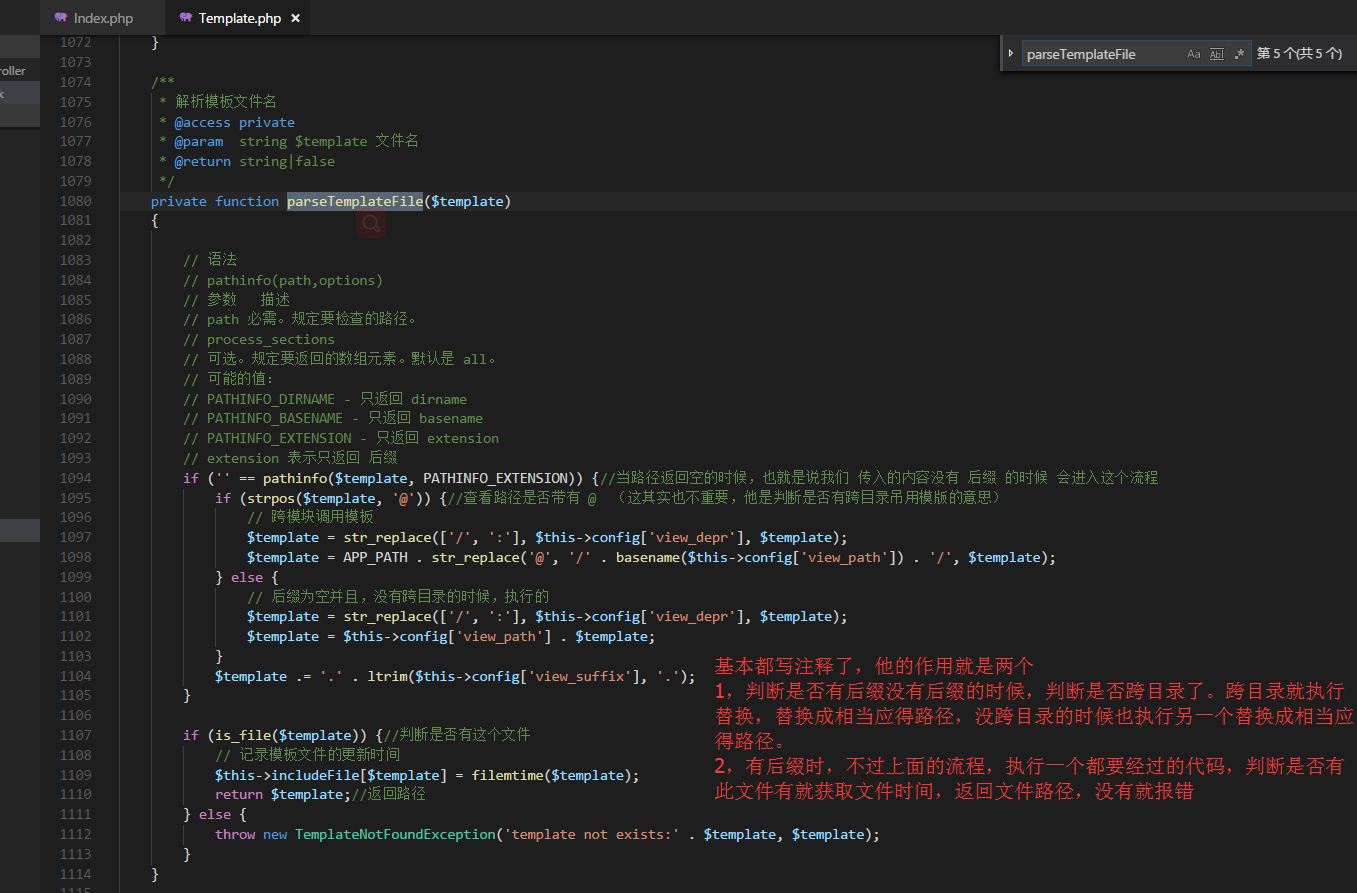
函数：fetch()

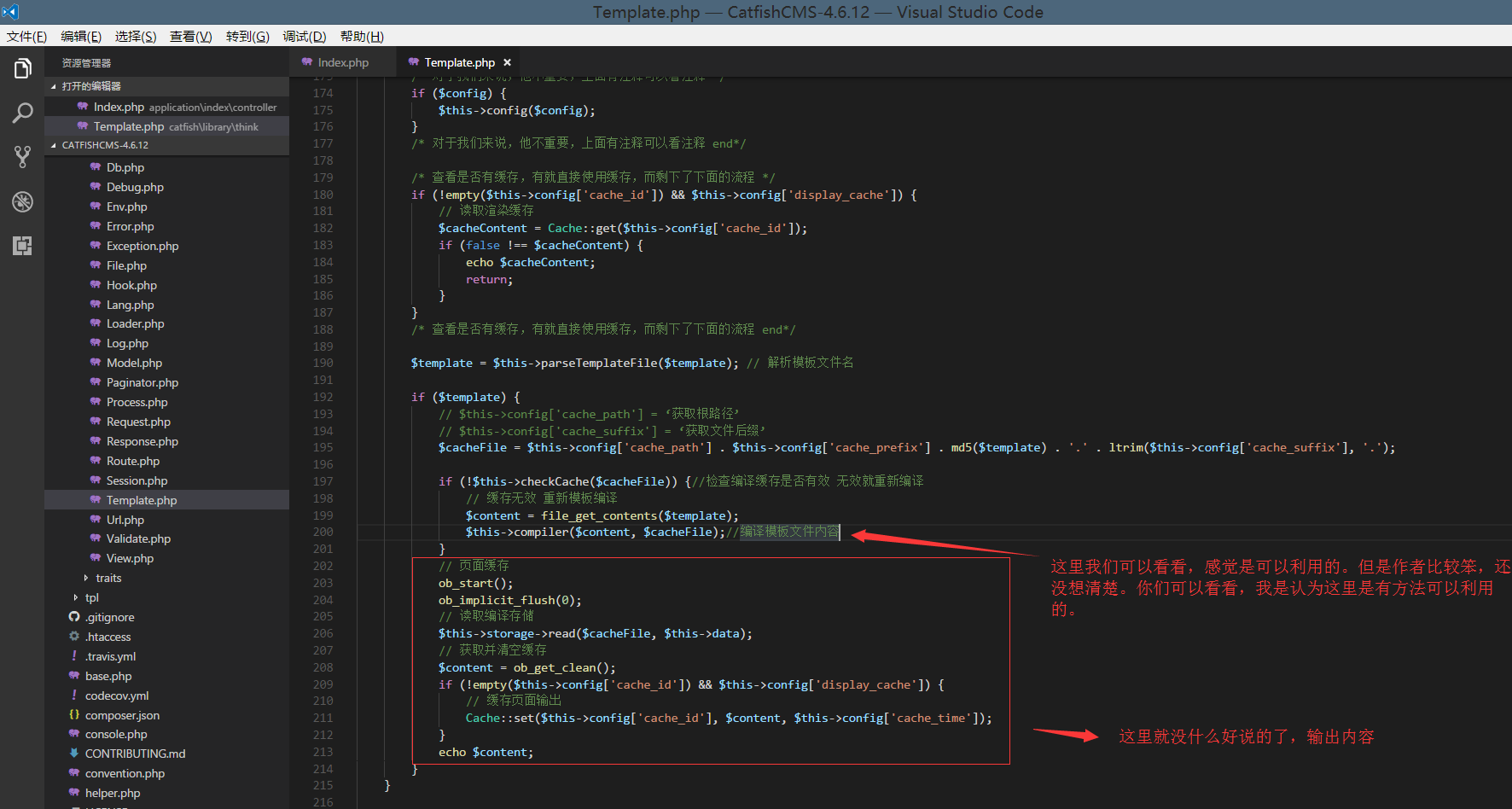












嗯，说完了。Fetch方法最后都会编译文件如何通过PHP输出，所以如果我们可以在他编译之前写入 恶意代码 那么就可以为所欲为。

**0x06初步介绍：**

在测试的过程中发现更新版本的时候作者添加一个参数 verification 可能是用来防治csrf。嗯，不得不说，作者安全意识提高了，用来防治csrf这的却是一个好思路，但是对于我们可以执行xss来说，verification 就显的苍白无力了，因为我们可以先获取 verification 然后在执行csrf 从而来绕过。

**0x07执行思路：**

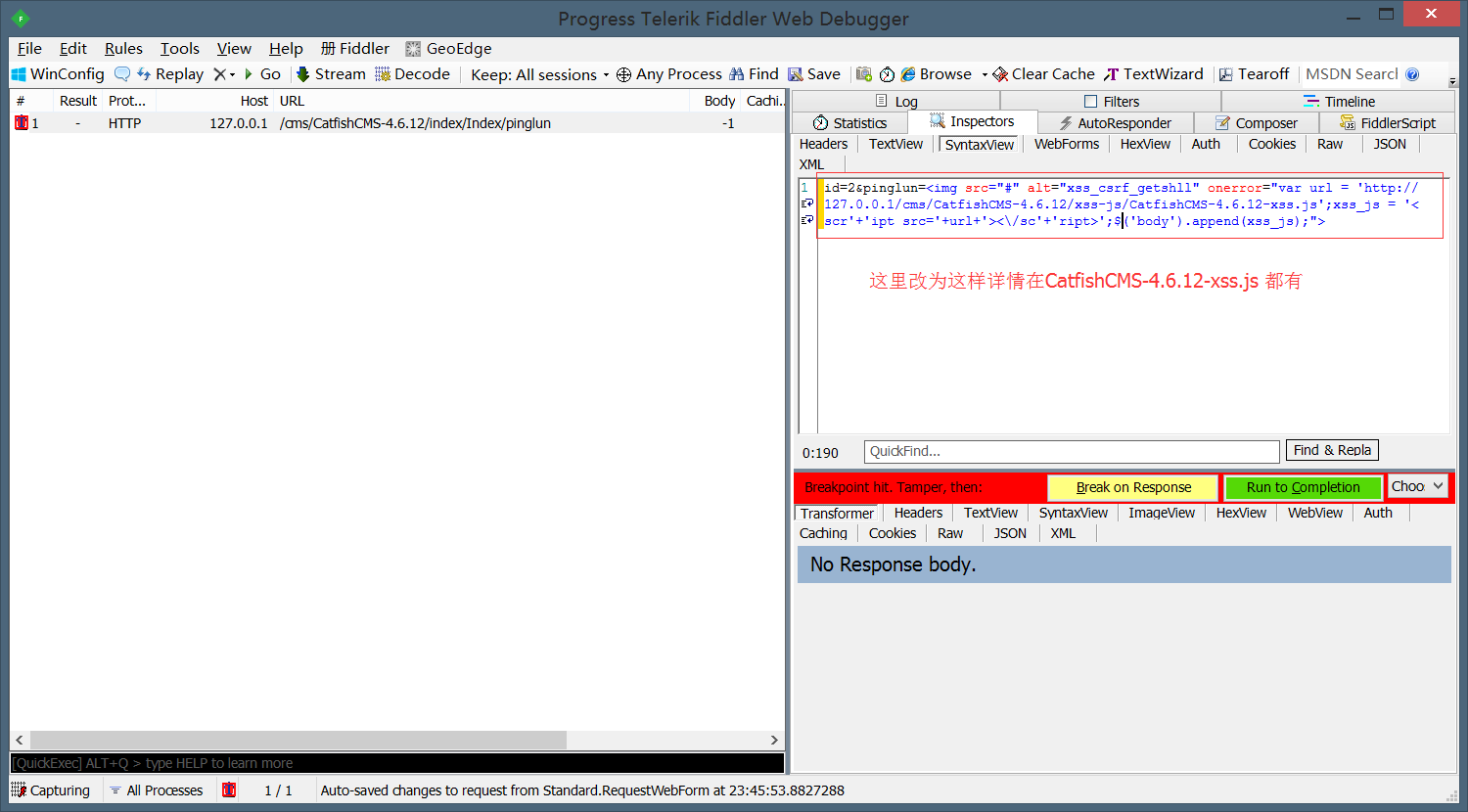
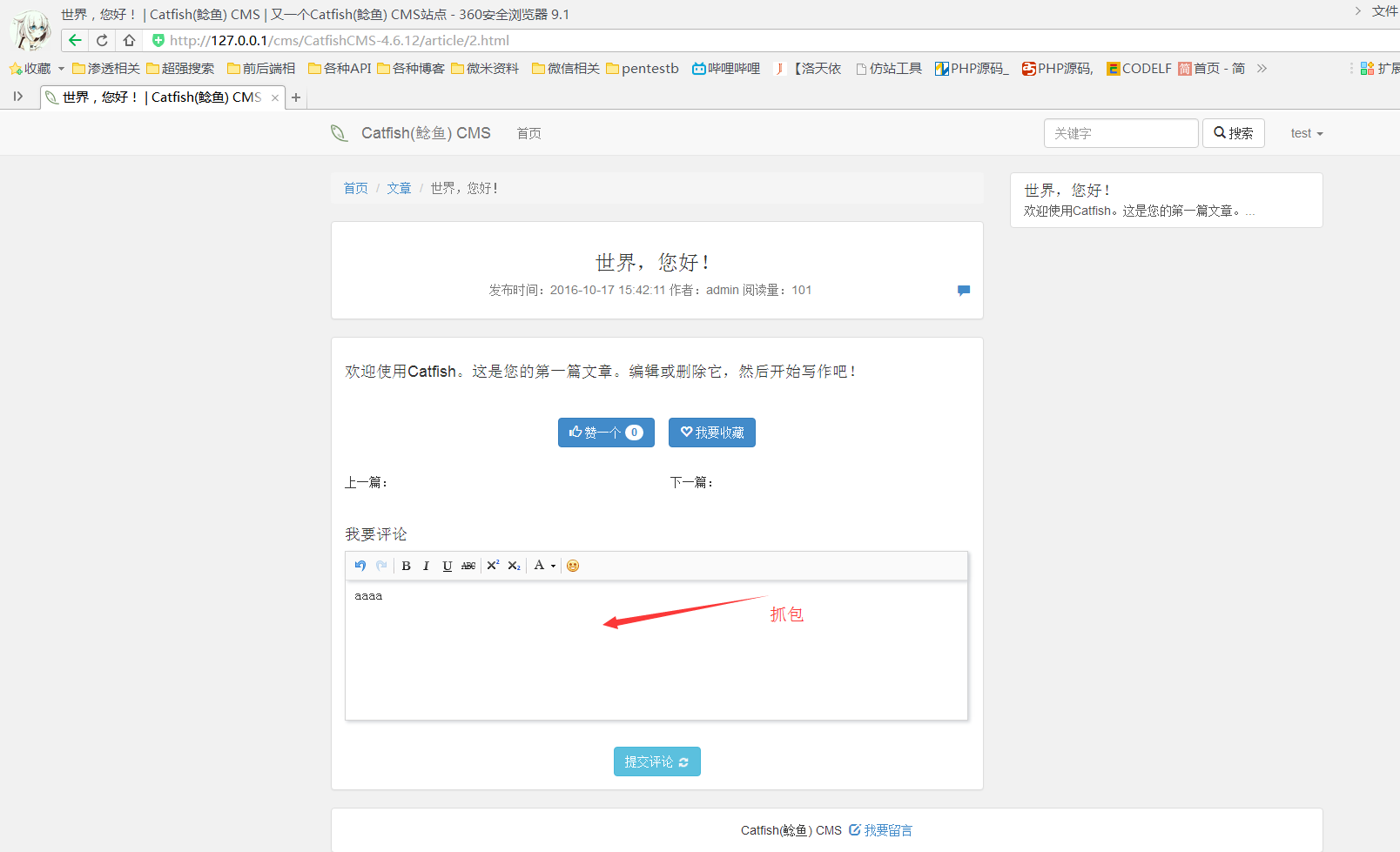
前台注册一个账户->注册一个图片马到网站中->评论处插入xss代码->等待无辜管理员登录网站->获取 verification(绕过检测)->管理员入口-页面管理-新建页面-csrf插入一条非法语句引起包含漏洞(用来包含前面的图片马)->包含漏洞执行代码->包含漏洞添加getshell\_code.php文件写入恶意代码->包含漏洞-将框架文件start.php添加一句话木马-包含数据库配置文件-连接数据库->删除我们前面的评论->删除我们csrf创建的页面->邮件通知我们->美滋滋

大概的想法就是这样了。实际的话，就让我们开始吧

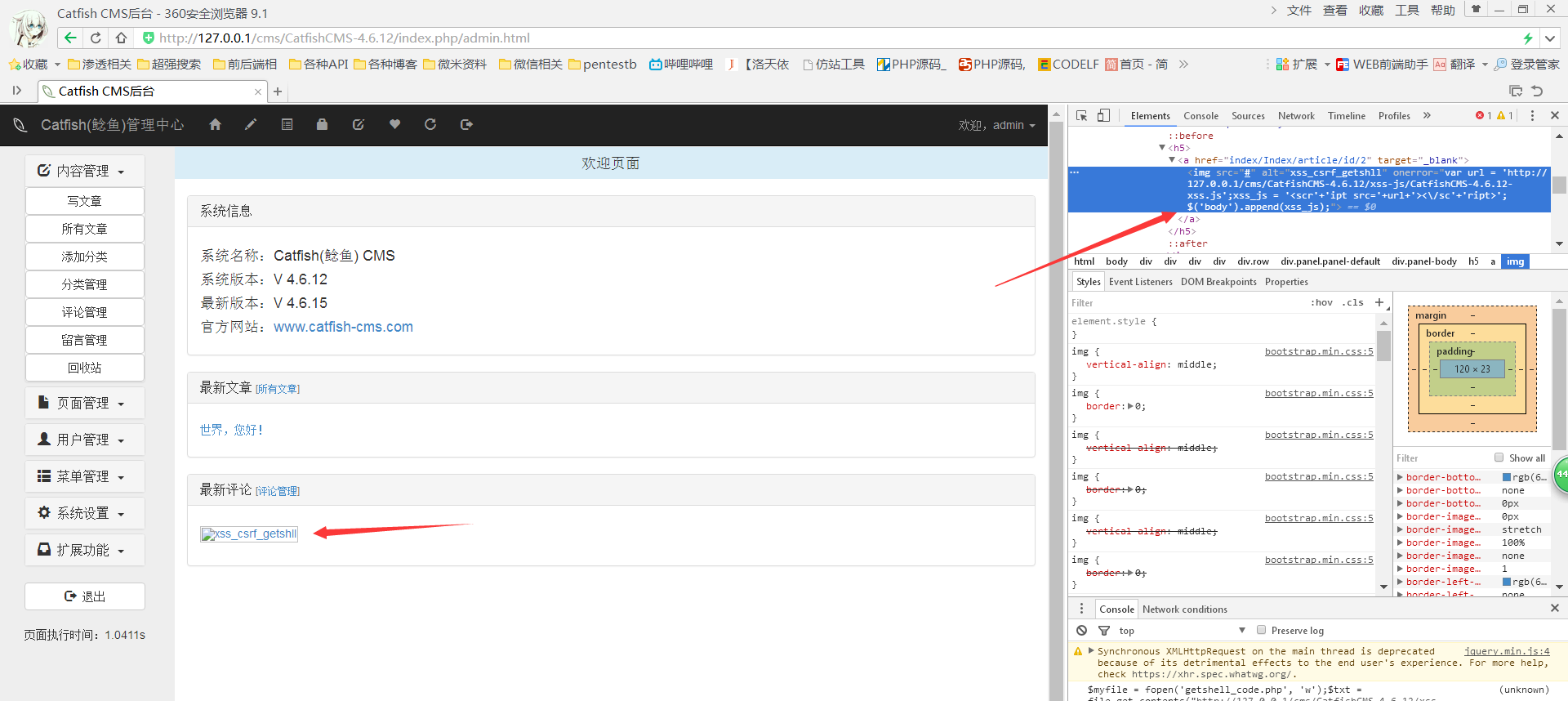
**0x08小小小小小小工具抒写：**

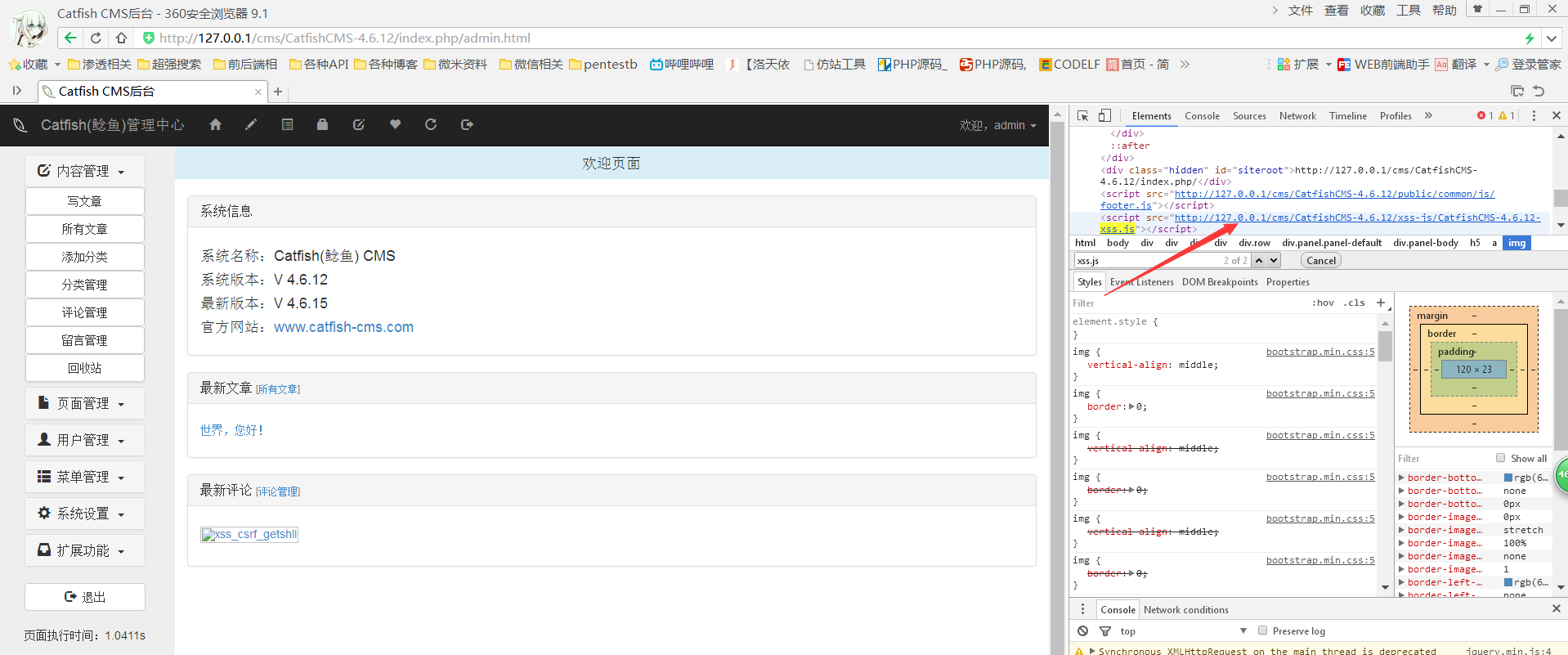
小小小小小工具已经写好了，详情看压缩包即可。

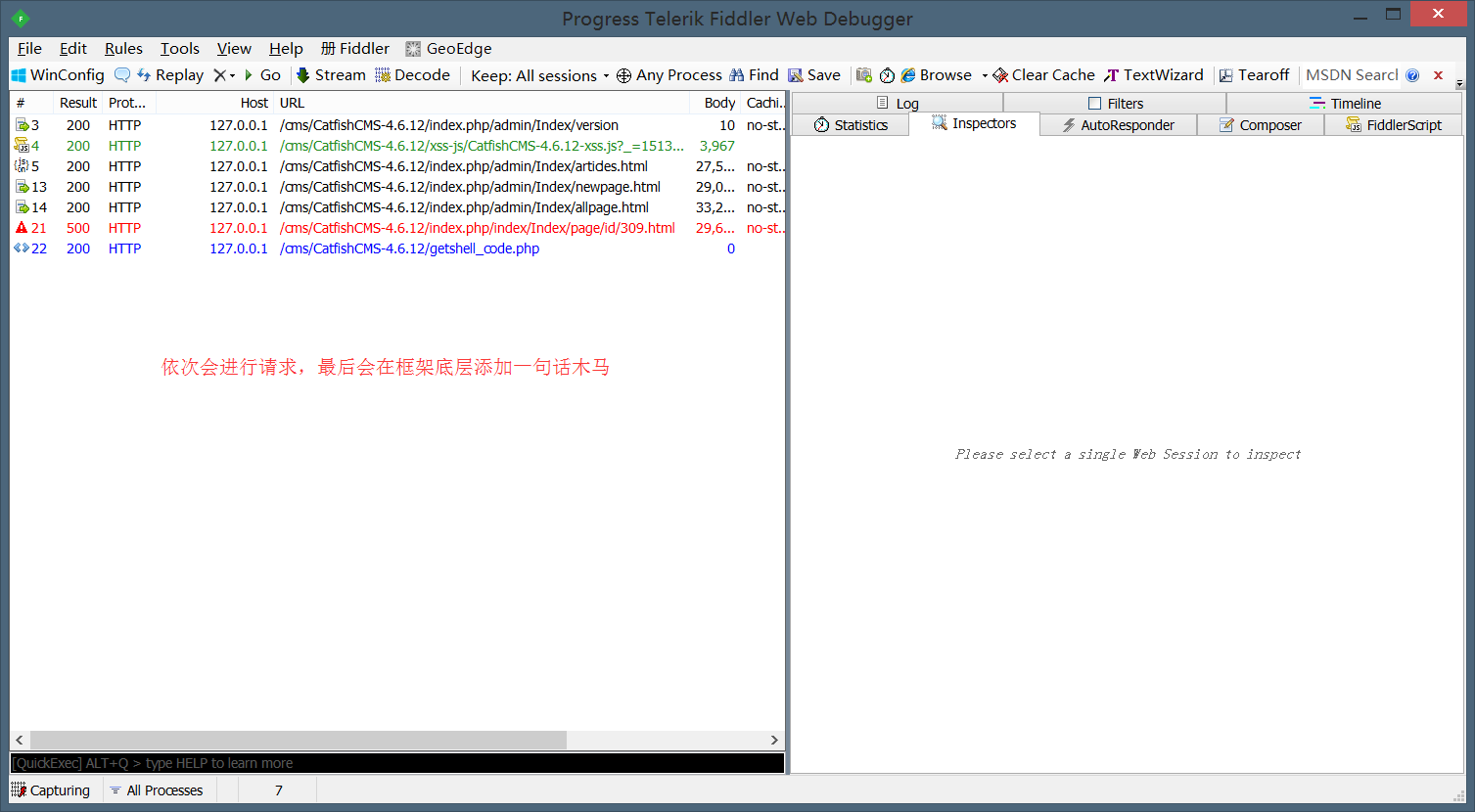
接下来试试效果。



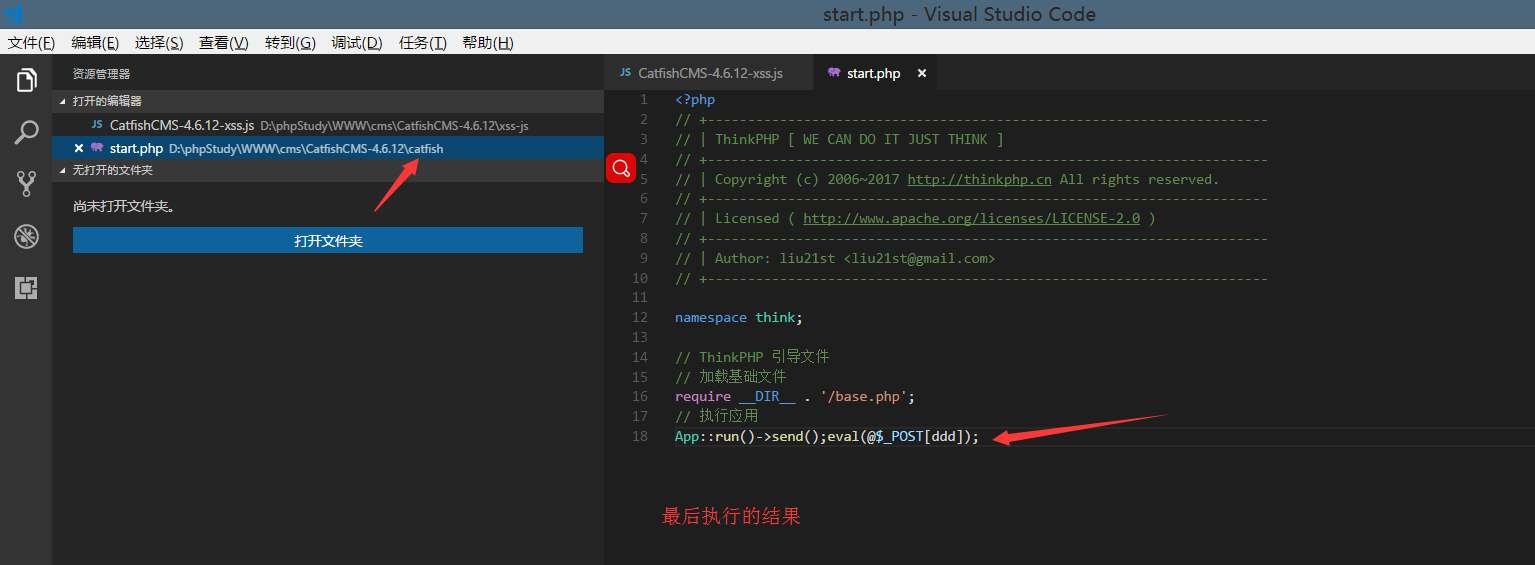
写好以后模拟管理员，进入后台











连接马子的操作，这个文件会在index.php 中给引入所以至今

<http://xxx.com/index>

POST

ddd = 你要执行的命令

