记一次曲折的 WAF 绕过 – 先知社区

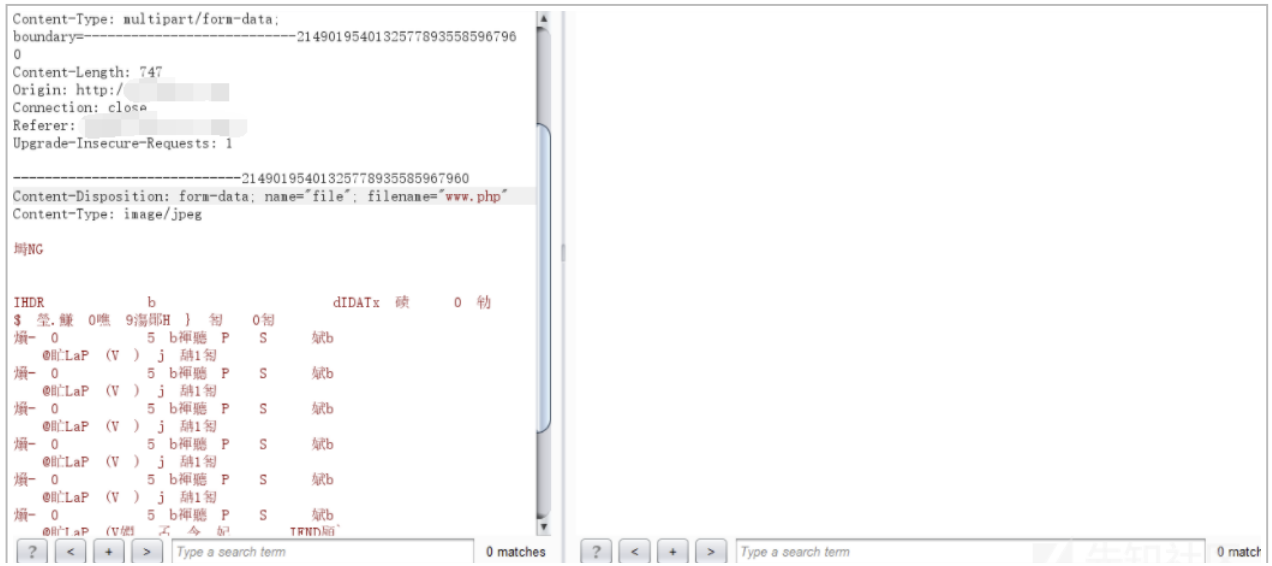
先知社区，先知安全技术社区

0x00：前言

做之前没想过有这么难

0x01：后缀绕过

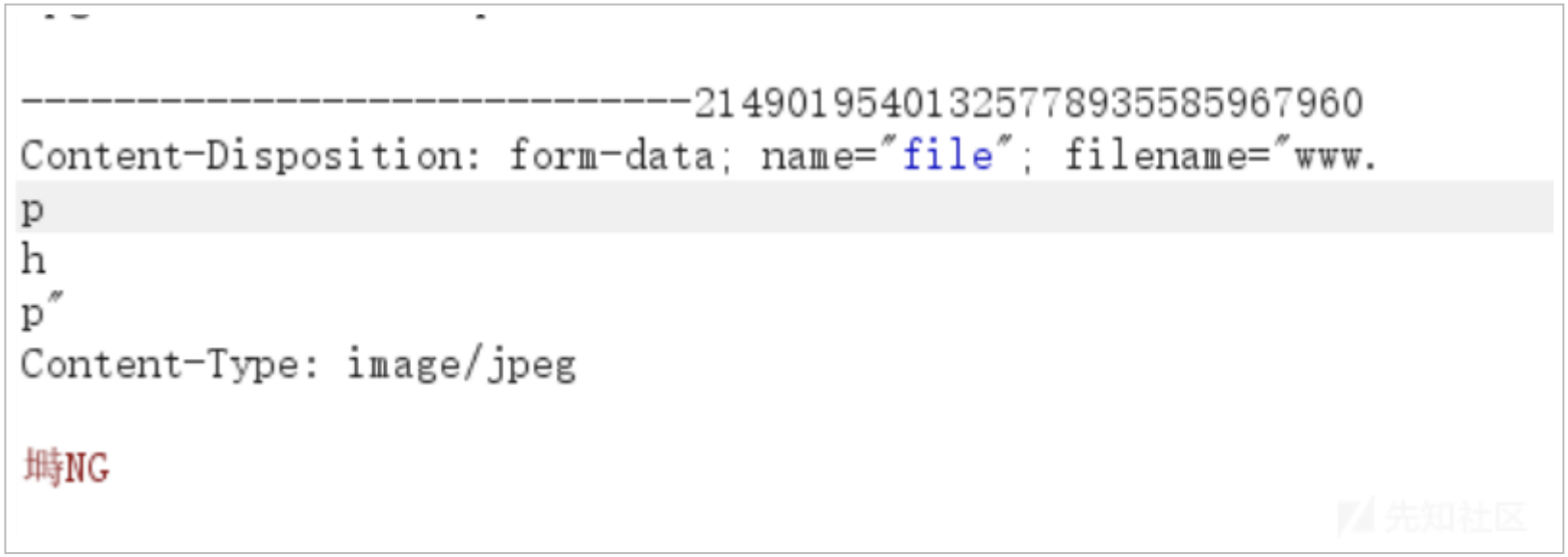
首先看一下 waf 咋工作的，当数据包匹配到 waf 规则后，数据包就会被丢弃掉，就像这样



(https://xzfile.aliyuncs.com/media/upload/picture/20211101095357-9368b4de-3ab6-1.png)

waf 是拦截后缀的，首先 fuzz 一波

换行



(https://xzfile.aliyuncs.com/media/upload/picture/20211101095425-a4420c92-3ab6-1.png)

失败

多个等于号

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

(https://xzfile.aliyuncs.com/media/upload/picture/20211101095449-b276c1b8-3ab6-1.png)

失败

单双引号替换

```
Upgrade-Insecure-Requests: 1

-----------------------------214901954013257789355859679 60
Content-Disposition: form-data; name="file"; filename='www.php
Content-Type: image/jpeg
```

(https://xzfile.aliyuncs.com/media/upload/picture/20211101095506-bc9e6cea-3ab6-1.png)

失败

去掉引号

```
-----------------------------214901954013257789355859679 60
```

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

(https://xzfile.aliyuncs.com/media/upload/picture/20211101095547-d4fb262a-3ab6-1.png)

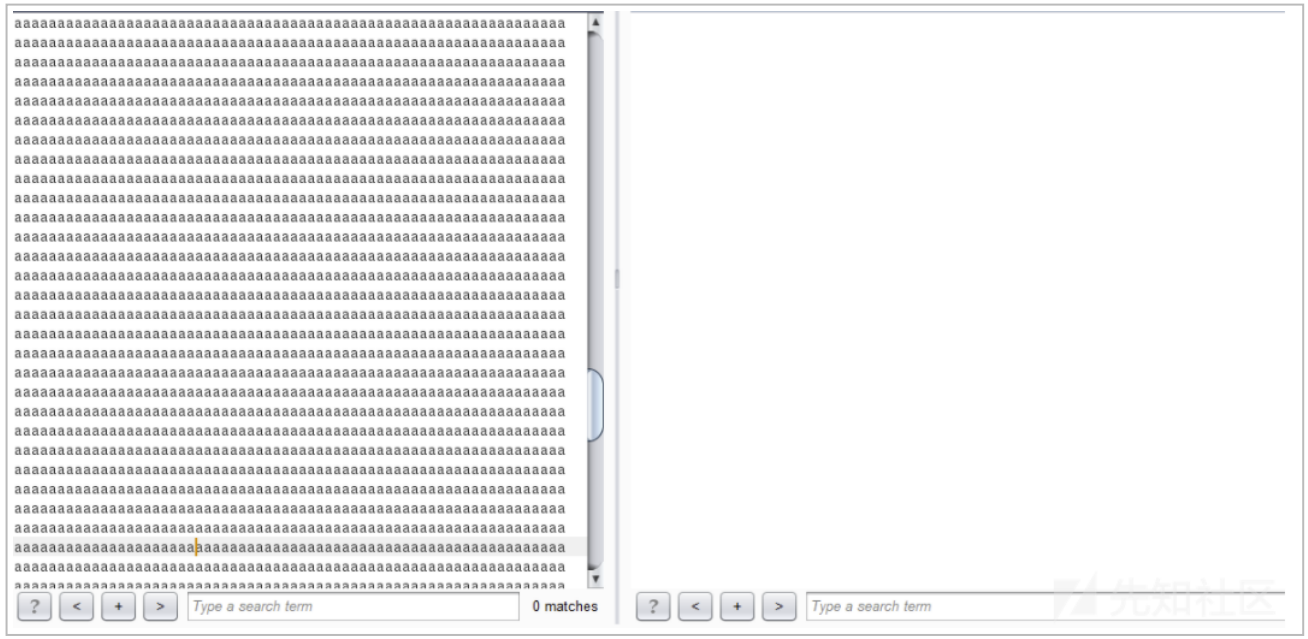
失败

溢出 Content-Disposition 字段

```
-----------------------------21490195401325778935585967960
Content-Disposition:aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa form-data;
name="file"; filename=www.php
Content-Type: image/jpeg
```

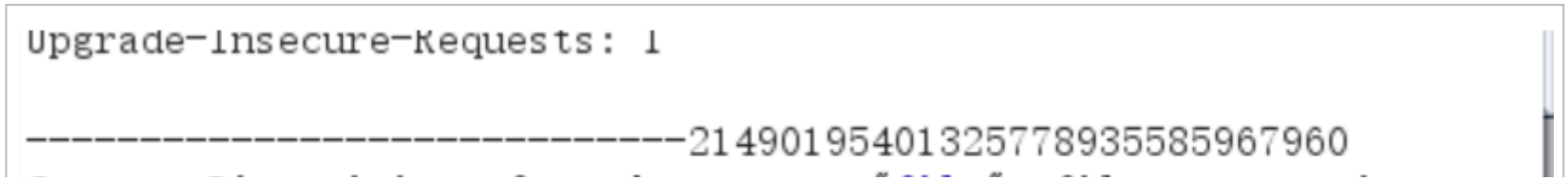
(https://xzfile.aliyuncs.com/media/upload/picture/20211101095633-f0c81124-3ab6-1.png)

失败，而且不清楚是因为服务器性能原因还是规则有这个，当此字段太长的时候（具体多长不清楚），正常上传图片数据包也会被丢弃。



(https://xzfile.aliyuncs.com/media/upload/picture/20211101095650-fa88607e-3ab6-1.png)

多个 Content-Disposition 字段



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

(https://xzfile.aliyuncs.com/media/upload/picture/20211101095705-035907ee-3ab7-1.png)

失败

畸形协议

```
POST /upload_file.php HTTP/1.9
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0)
Gecko/20100101 Firefox/93.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,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
Content-Type: multipart/form-data;
boundary=---------------------------214901954013257789355859679
60
```

(https://xzfile.aliyuncs.com/media/upload/picture/20211101095806-282a2274-3ab7-1.png)

失败

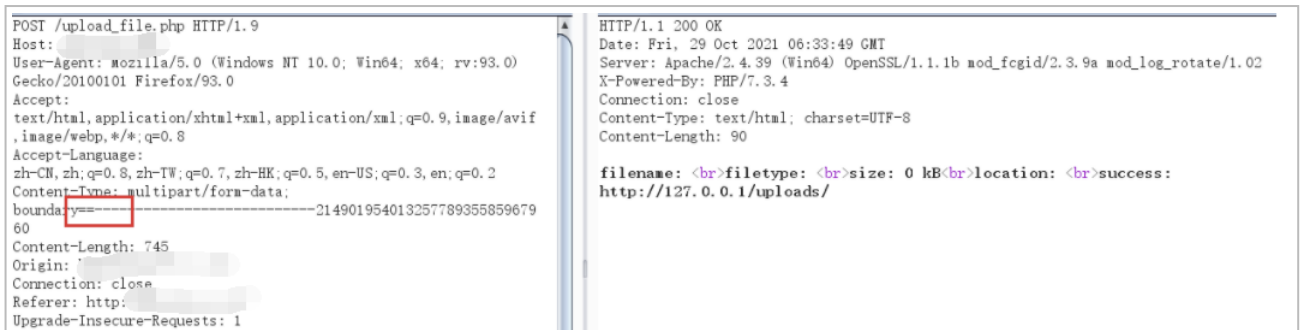
失败

boundary 前加减空格

```
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,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
Content-Type: multipart/form-data;
boundary===--------------------------214901954013257789355859679
60
Content-Length: 745
```

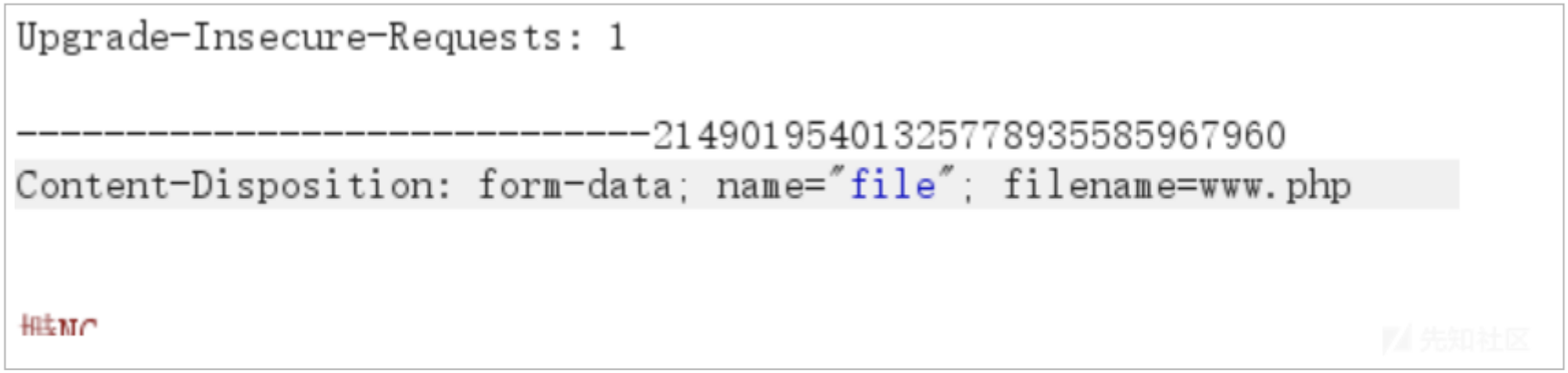
(https://xzfile.aliyuncs.com/media/upload/picture/20211101095829-3586dcbe-3ab7-1.png)

失败，而且操作这边服务器会不识别上传，导致正常文件传不上去



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100200-b39f9014-3ab7-1.png)

删除 Content-Type: image/jpeg



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100214-bb769a94-3ab7-1.png)

失败

溢出文件名

```
Content-Disposition: form-data; name="file";
filename=wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww.php


坶NG
```

(https://xzfile.aliyuncs.com/media/upload/picture/20211101100233-c6d7d9fc-3ab7-1.png)

失败，而且这里也有限制，太长也会导致正常上传失效。

Accept-Encoding：

Accept-Encoding: gzip

Accept-Encoding: compress

Accept-Encoding: deflate

Accept-Encoding: br

Accept-Encoding: identity

Accept-Encoding: *

均失败

分块传输

```
POST /upload_file.php HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0)
Gecko/20100101 Firefox/93.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,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
Content-Type: multipart/form-data;
boundary=---------------------------214901954013257789355859679 6
```

```
0
Content-Length: 7762
Origin: h
Connection: close
Referer: http:
Upgrade-Insecure-Requests: 1
Transfer-Encoding: chunked

2;1H2Rmbd6
--
3;hUBLkEhvDDmXGr0
---
1;onYd1xFt3P
-
1;swiePzVdxfHgh5U5ePN7kFQTf
-
1;eizpyKLJPnzx
-
1;8QSolF1W2rT3VLF2zRsuR0L
```


(https://xzfile.aliyuncs.com/media/upload/picture/20211101100327-e726f0f8-3ab7-1.png)

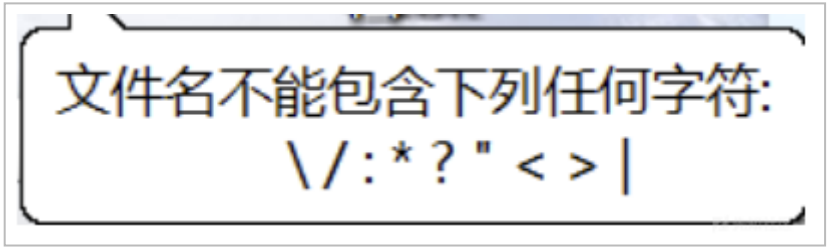
失败 似乎 waf 拦截的就是 Transfer-Encoding: chunked ，但是去掉 chunked 会请求异常

等等等后面又试了一堆，均失败

柳暗花明

后面看了一眼服务器 是 windows 的，尝试用 windows 文件命名规范来绕过，

众所周知，win 的文件名是不能包含以下字符的



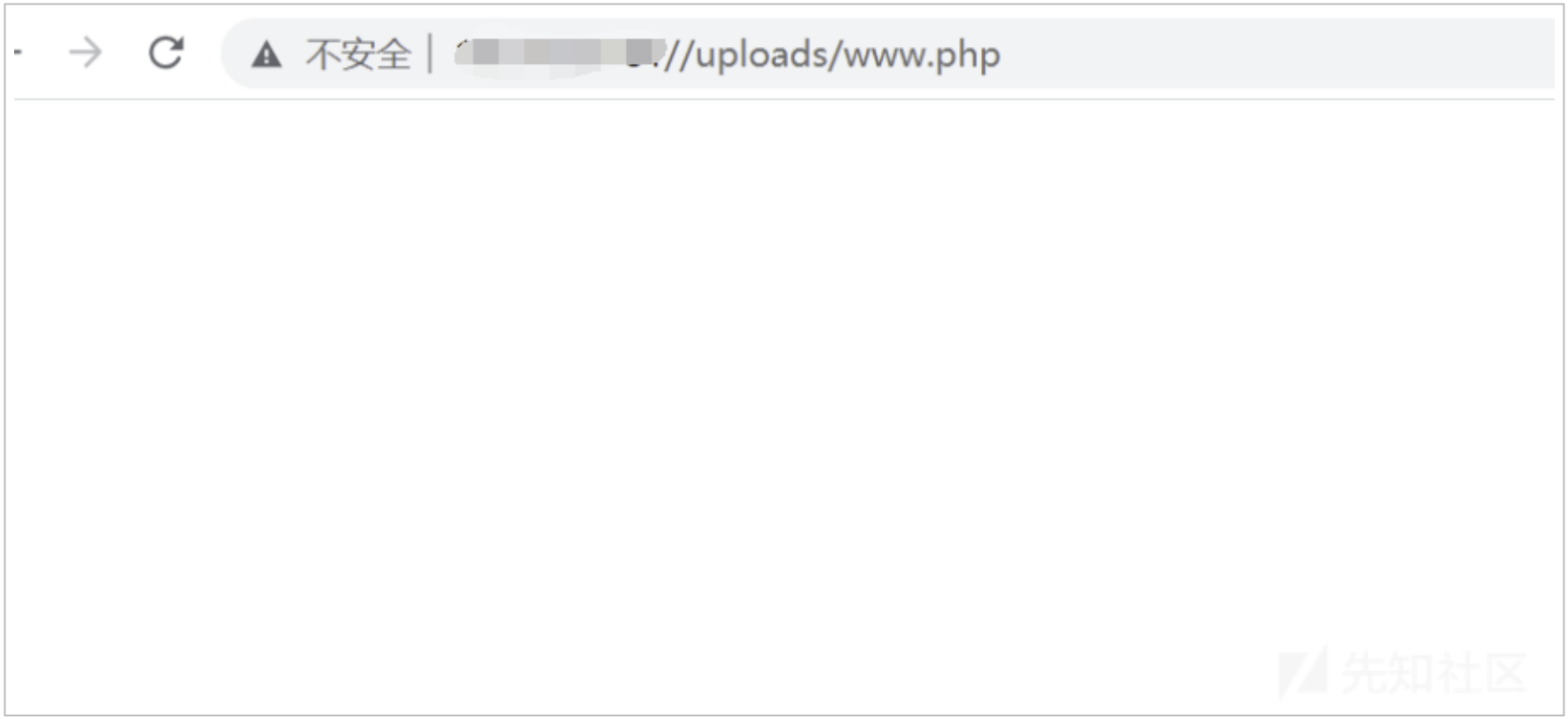
(https://xzfile.aliyuncs.com/media/upload/picture/20211101100353-f7035246-3ab7-1.png)
但是上传的时候我们可以构造，尝试使用斜杠绕过

(https://xzfile.aliyuncs.com/media/upload/picture/20211101100417-05558756-3ab8-1.png)
结果有点出乎意料，最后到服务器的居然是. jpg, 正反斜杠都这样

换一个符号，尝试星号，被拦，都试了一遍后，发现只有冒号可以

(https://xzfile.aliyuncs.com/media/upload/picture/20211101100444-154a1adc-3ab8-1.png)

访问一下看看是否存在 www.php



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100501-1f0e178a-3ab8-1.png)

文件确实传上去了，可是问题又来了，没写入内容...

后来才知道，是是会将文件内容置空，一时间又没了头绪

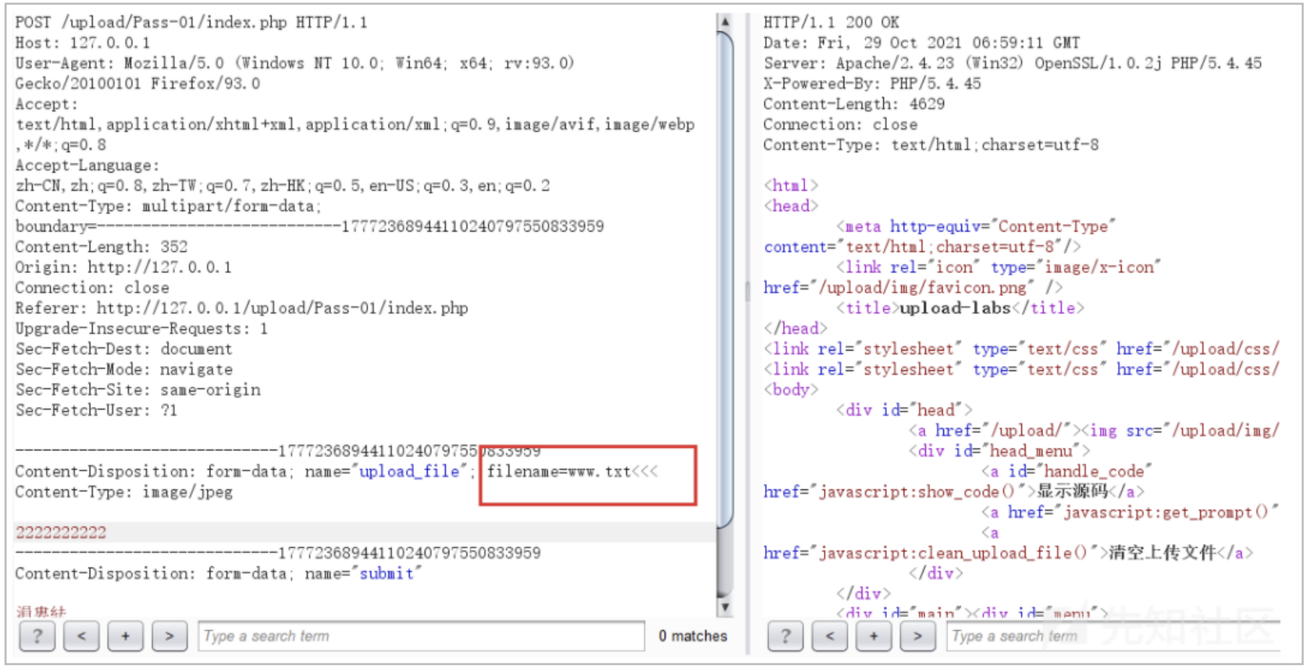
姑且后缀名绕过了吧。

0x02：内容绕过

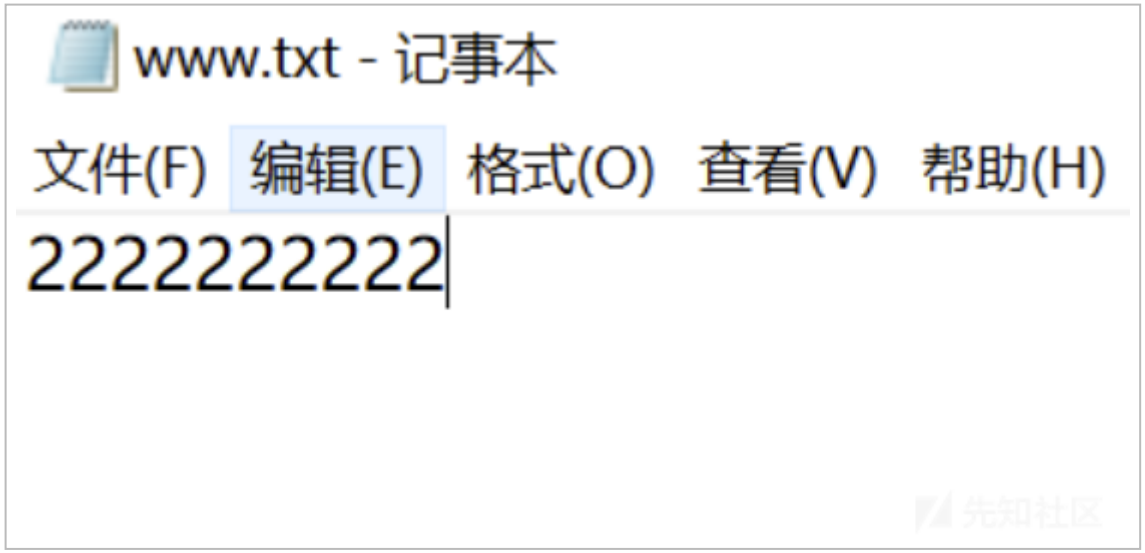
只传个 0kb 和不传不是一样吗，所以还是要把数据写进去，

怎么写呢？起初我是不知道的，旁边的好兄弟说三个左尖括号可以写入文件

类似这样



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100548-3b04fa9e-3ab8-1.png)

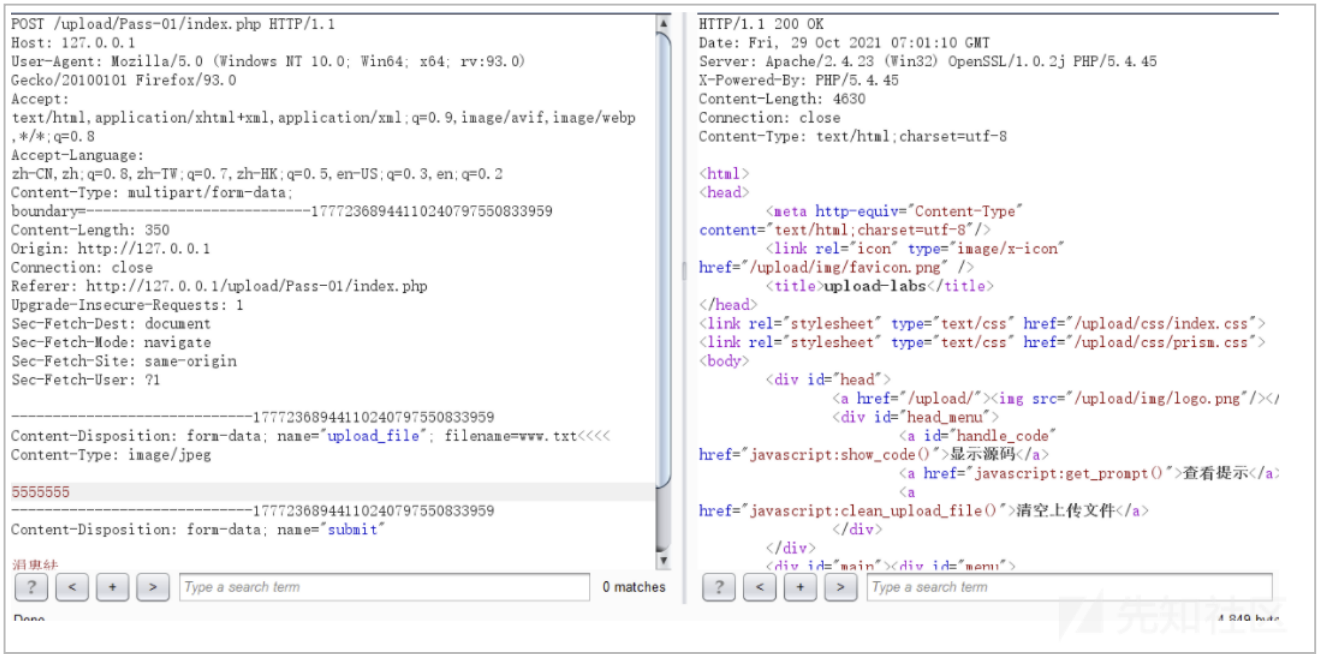


(https://xzfile.aliyuncs.com/media/upload/picture/20211101100554-3edc7fd4-3ab8-1.png)
但是 也被 waf 加入规则了

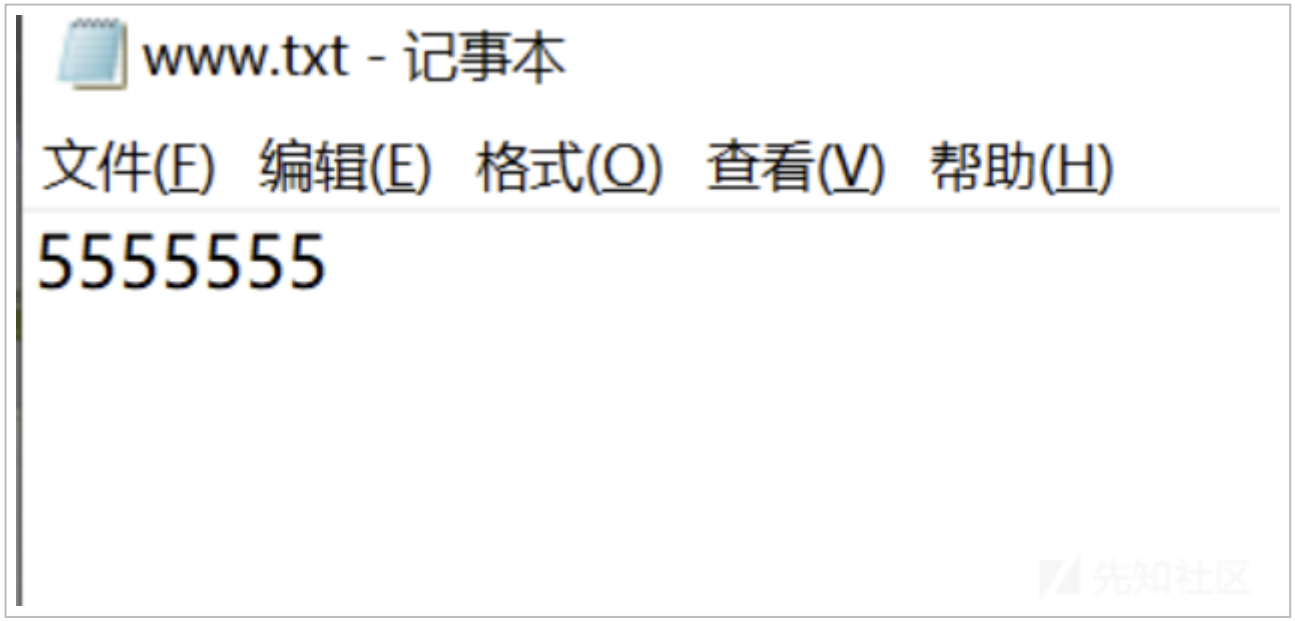
.....

天无绝人之路，三个不行 我用四个

没想到四个也能写 笑了 都不知道为什么



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100612-49667f86-3ab8-1.png)



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100617-4cdc574e-3ab8-1.png)

但是问题又来了 文件名咋整，www.php 会被拦截，加个冒号又会将文件置空，似乎陷入了死循环，一顿瞎操作后，发现这样居然写进去了，虽然也不知道为什么，可能是什么奇奇怪怪的正则机制?

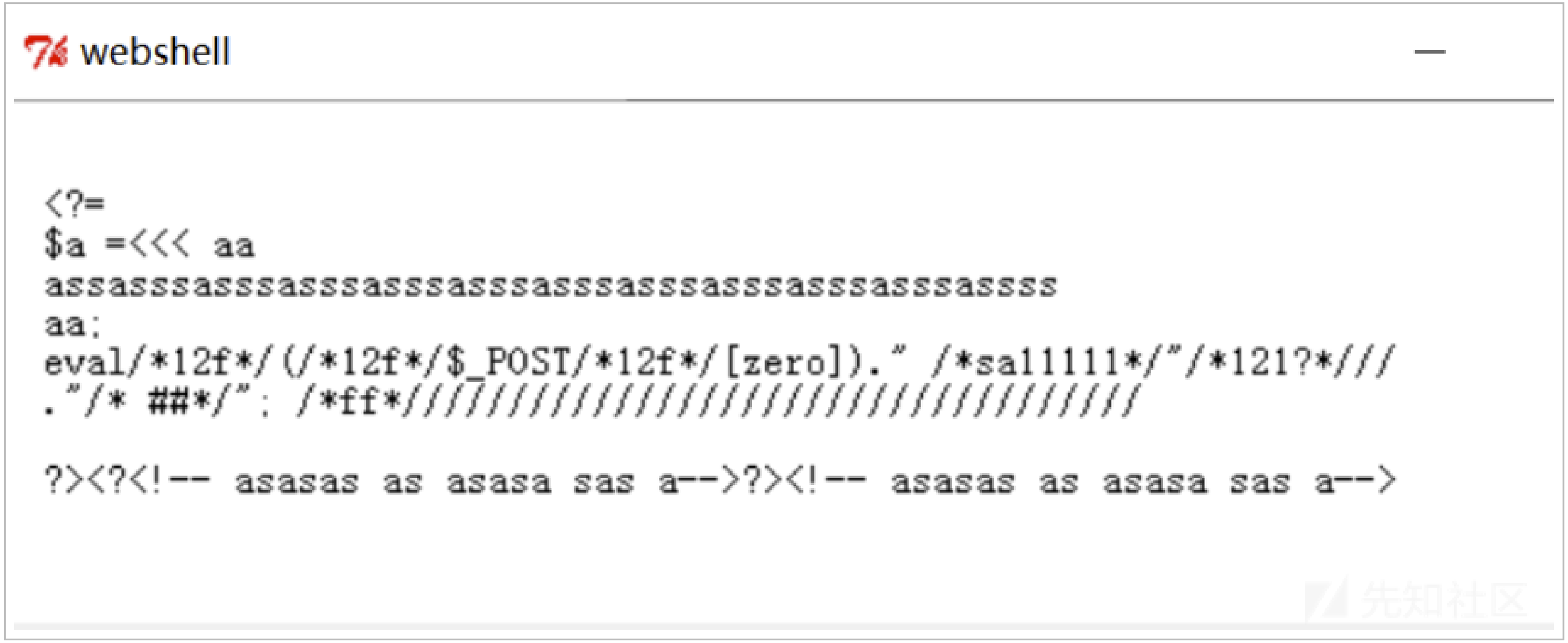
(https://xzfile.aliyuncs.com/media/upload/picture/20211101100651-6090b42e-3ab8-1.png)

(https://xzfile.aliyuncs.com/media/upload/picture/20211101100703-67d061e4-3ab8-1.png)

既然能写文件了，那我本来以为就简简单单了，没想到噩梦才是刚刚开始

首先我之前整的一堆花里胡哨的马一个都没过去，唯一一个能过去的，之前绕过的马还不能运行（这是个坑，后面会讲），
虽然这个马在我本地是可以运行的，可能这就是玄学吧哈哈

虽然这个马在我本地是可以运行的，可能这就是玄学吧哈哈。

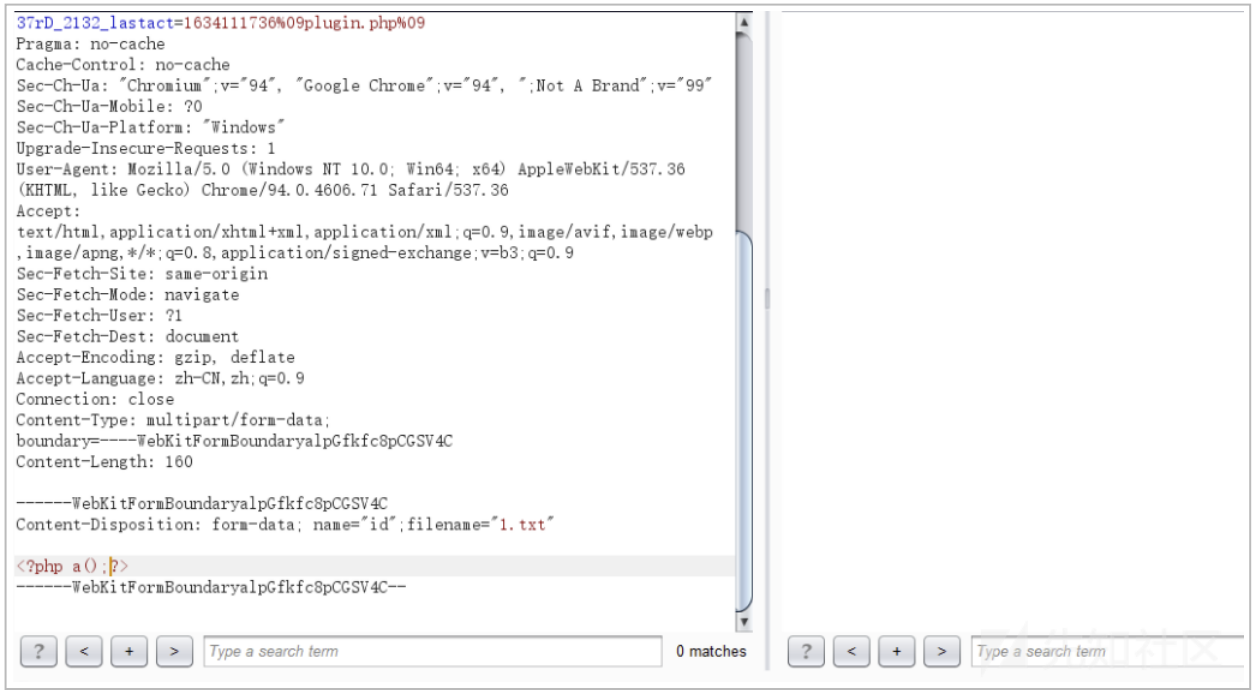


(https://xzfile.aliyuncs.com/media/upload/picture/20211101100720-71e14248-3ab8-1.png)

(https://xzfile.aliyuncs.com/media/upload/picture/20211101100755-87108fde-3ab8-1.png)

首先正常的变量他就过不去，然后 <?php 标签和某些场景混合时也过不去，成对的括号它也拦截，总之就是变态

举个例子：

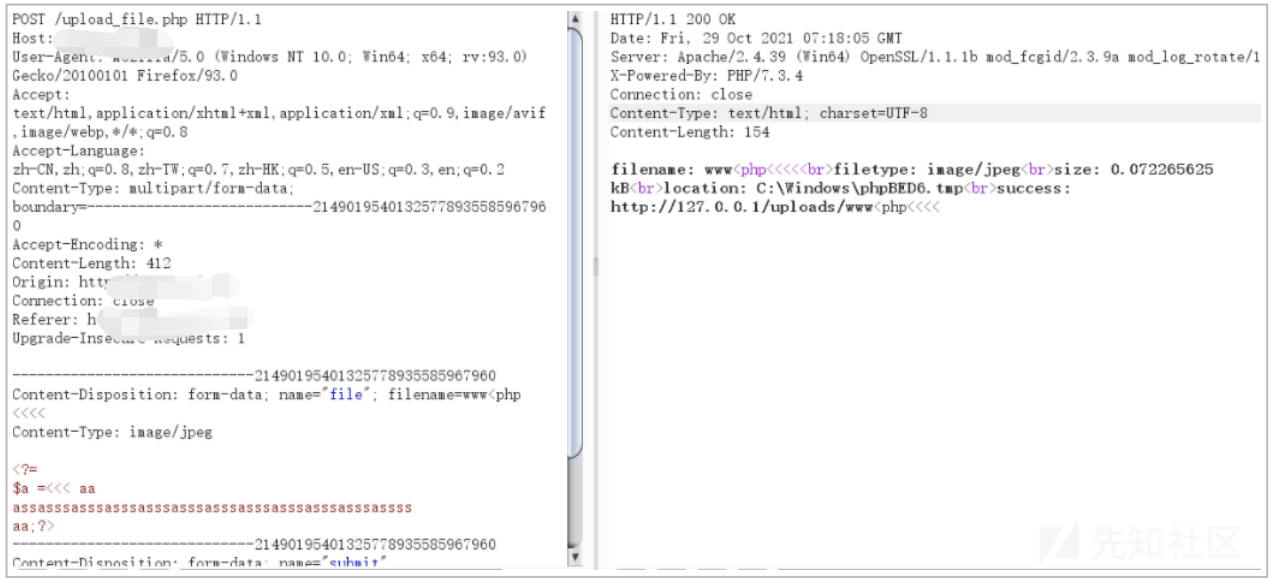


(https://xzfile.aliyuncs.com/media/upload/picture/20211101100827-9a4e23fe-3ab8-1.png)
所以才有了上面那个奇怪的马

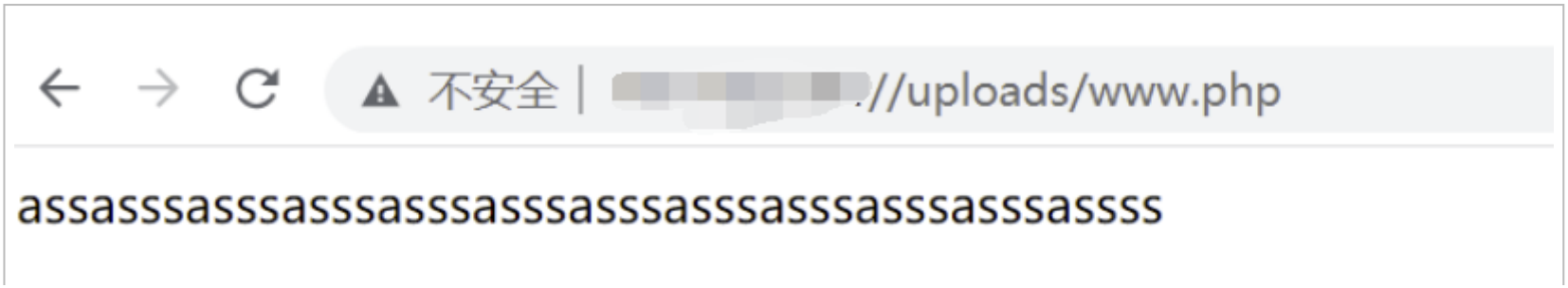
内容绕过也是在这个基础上去过去的，本来我准备直接传这个马梭哈，但是传是传上去了，执行不了，会一直等待，但是这个等待又不是被 waf 拦截了，后面才知道，这是因为 php 内容报错了，导致不能正常运行，就会一直卡在那里，

那为什么会报错呢，fuzz 了一波后才发现，这个服务器上必须闭合尖括号，要不然就会直接炸，明明是同样的 php 版本，

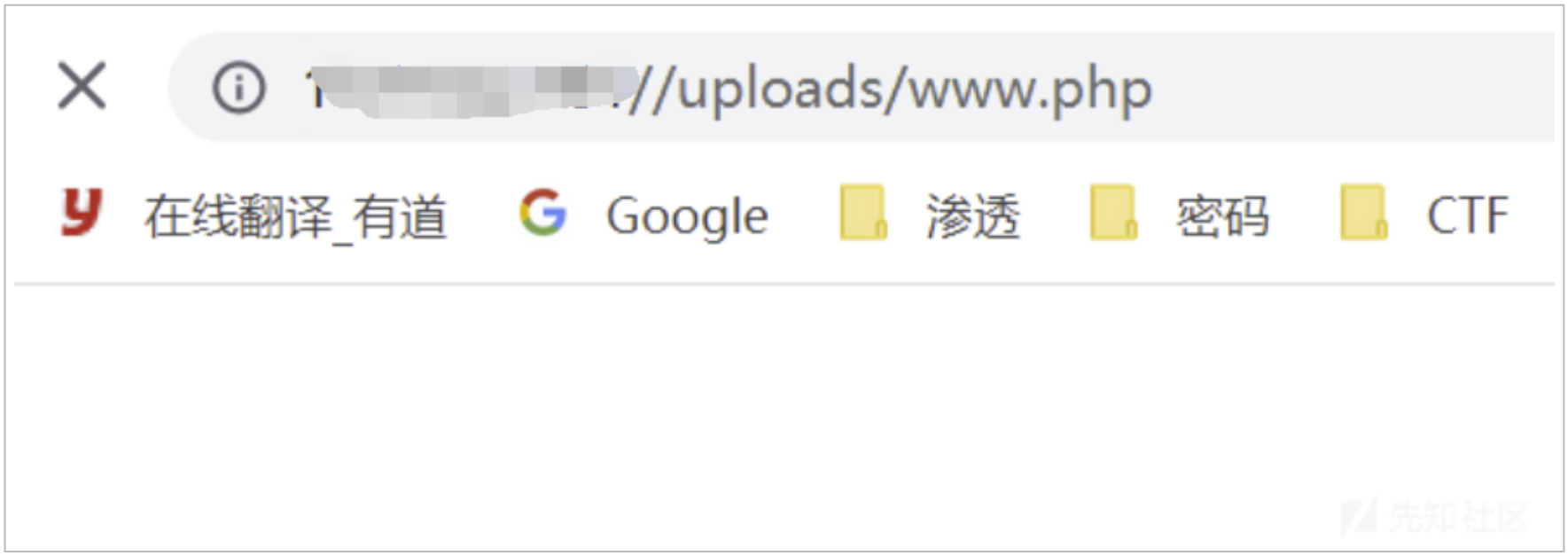
也不知道为什么会这样，只传递第一个标签没问题，



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100858-acbe0bda-3ab8-1.png)



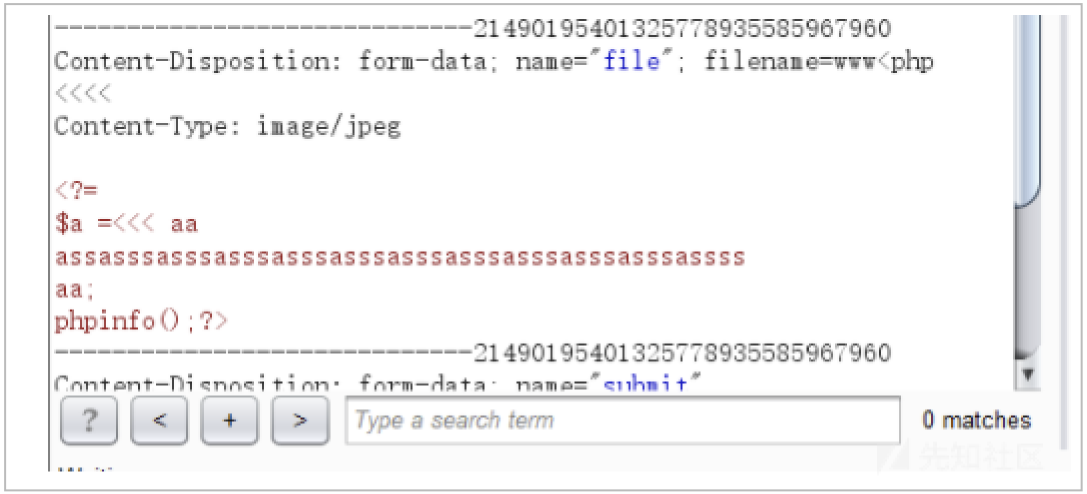
(https://xzfile.aliyuncs.com/media/upload/picture/20211101100909-b33ab3dc-3ab8-1.png)
少个闭合就炸：



(https://xzfile.aliyuncs.com/media/upload/picture/20211101100925-bc81b59e-3ab8-1.png)
但是末尾的两个标签也不能去掉，是用来混淆的，去掉直接 waf 都过不去

场面陷入了胶着，此时将一个完整的马传上去似乎不太现实了，一步一步来吧，先尝试能不能执行命令

之前讲过，规则是不允许成对括号出现的，所以连 phpinfo 都执行不了

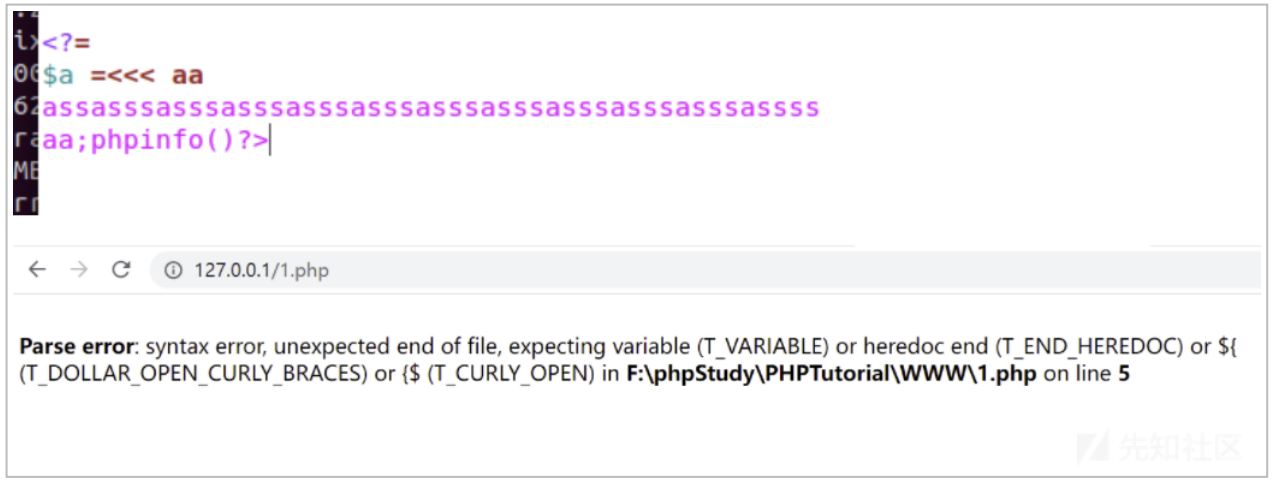


(https://xzfile.aliyuncs.com/media/upload/picture/20211101100950-cb89c2a2-3ab8-1.png)

真的执行不了吗?

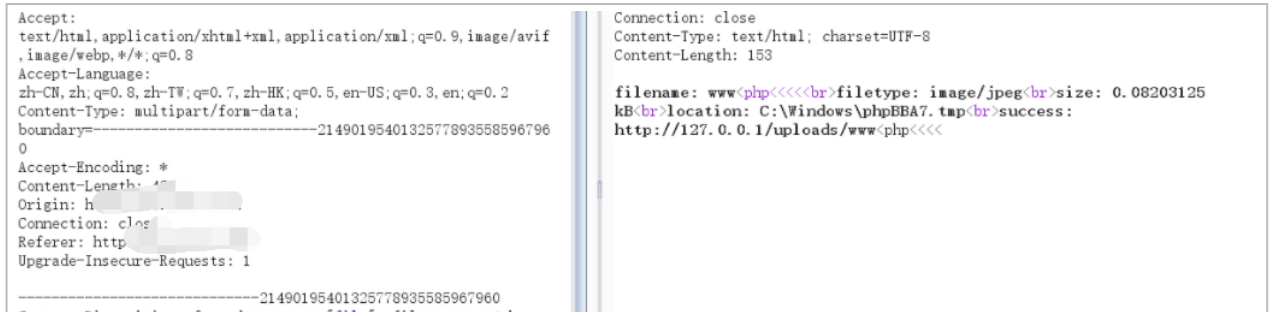
```
$a =<<< aa
assasssasssasssasssasssasssasssasssasssassss
aa;
```

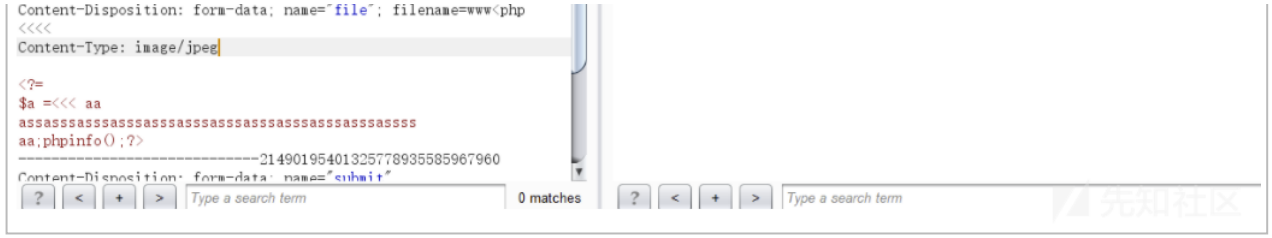
这种赋值手法是 php 的一种特性，用于解决字符串中既有单引号又有双引号这种特殊情况，aa 名称没意义，起什么都可以，在某些 php 版本中，末尾的 aa 后不能加其他语句，否则会报错。



(https://xzfile.aliyuncs.com/media/upload/picture/20211101101029-e28047c4-3ab8-1.png)

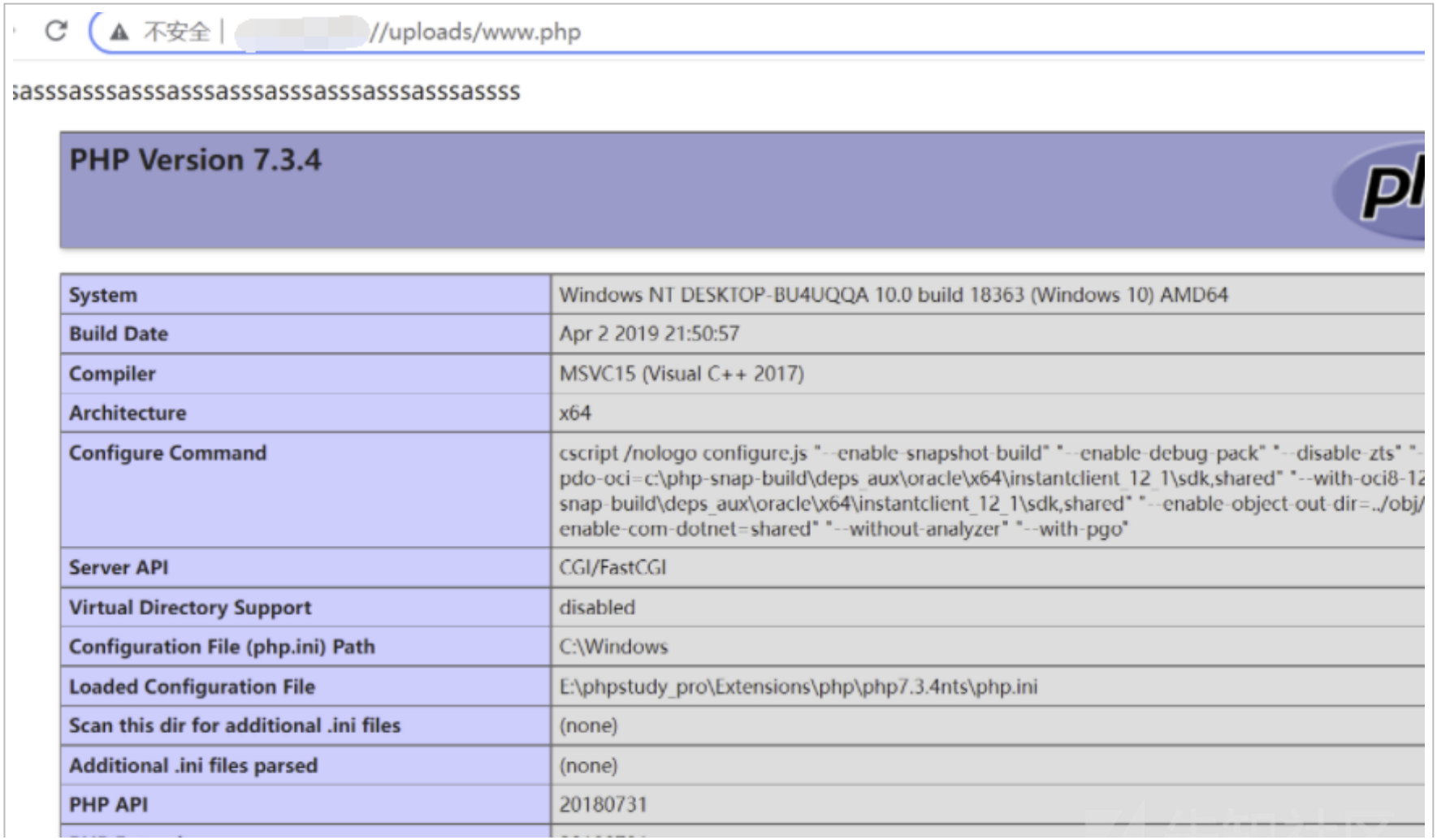
因为会报错，所以 waf 不会拦截，所以在 aa; 后是可以添加 php 代码的





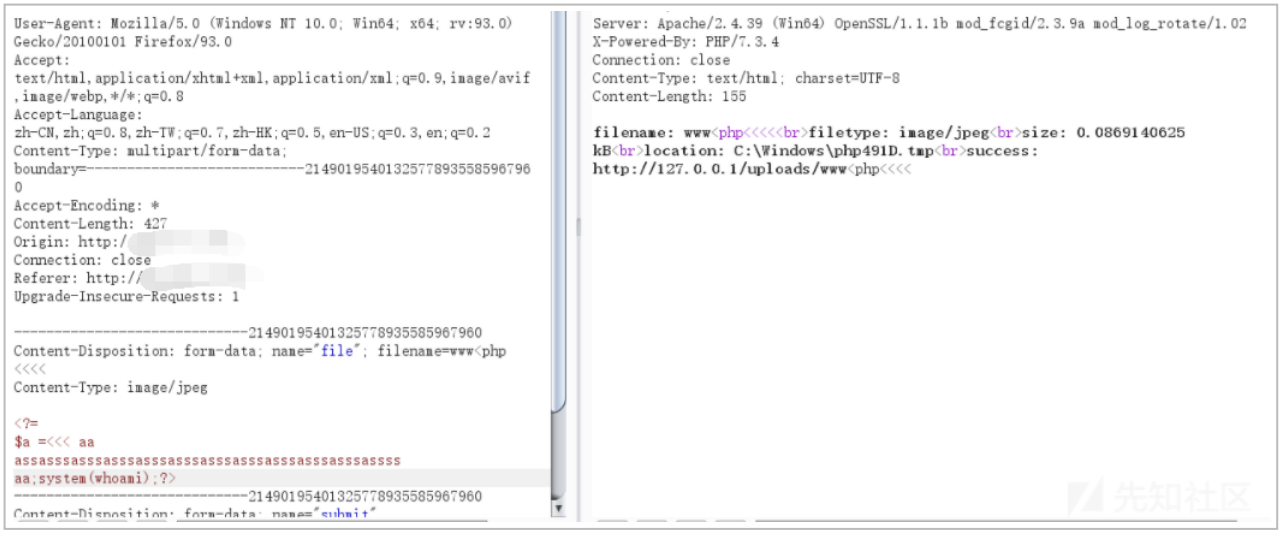
(https://xzfile.aliyuncs.com/media/upload/picture/20211101101051-eff769fa-3ab8-1.png)

好巧不巧，这服务器可以执行



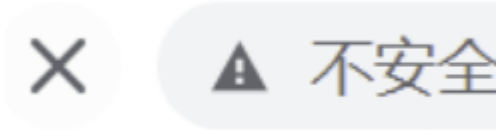
(https://xzfile.aliyuncs.com/media/upload/picture/20211101101110-fb1c64a2-3ab8-1.png)

既然能执行 phpinfo，那 system() 肯定也行



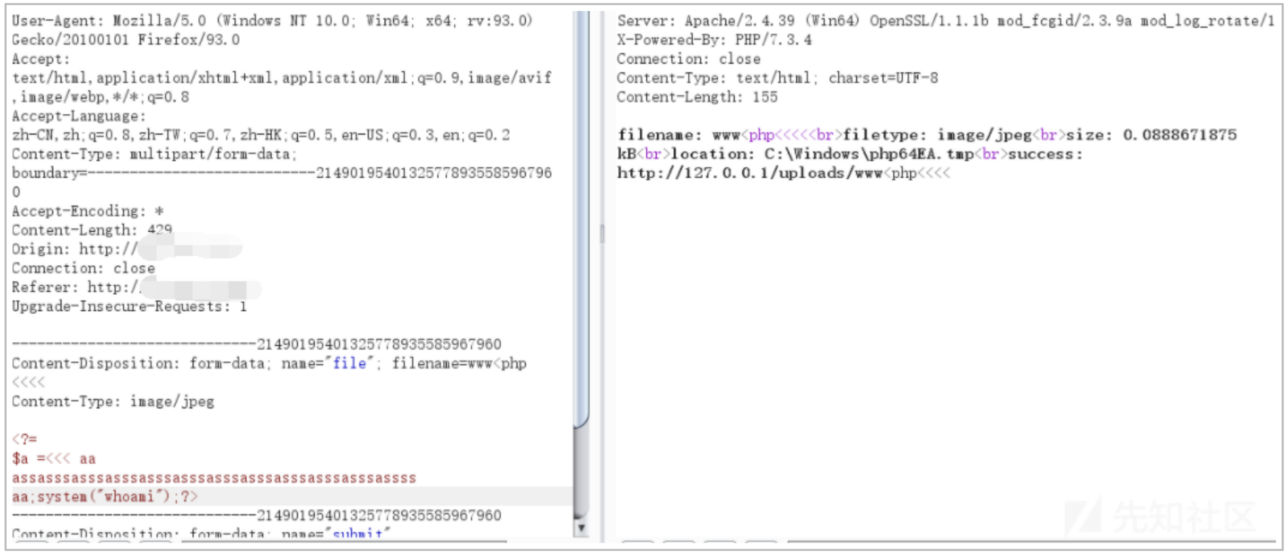
(https://xzfile.aliyuncs.com/media/upload/picture/20211101101136-0ada99fe-3ab9-1.png)

传完后又转起来了

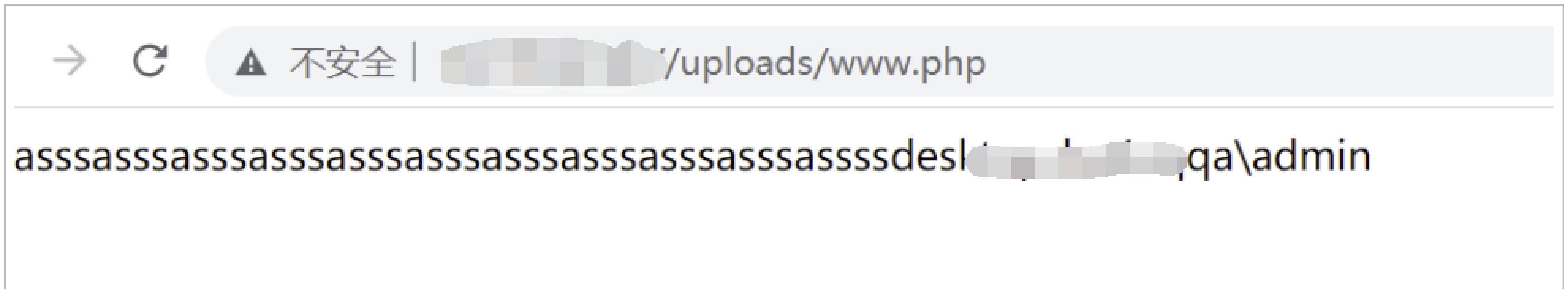


(https://xzfile.aliyuncs.com/media/upload/picture/20211101101148-11ba43c8-3ab9-1.png)

这里又一个坑，因为 system 中可以用双引号也可以不用，用双引号的时候一般是类似 ls -l 这种有空格的情况，但是 whoami 这种的是不需要引号的，但是这里访问一直转明显是语法错误，那我加上就是了。



(https://xzfile.aliyuncs.com/media/upload/picture/20211101101207-1cfd91ae-3ab9-1.png)



(https://xzfile.aliyuncs.com/media/upload/picture/20211101101233-2c9e7c40-3ab9-1.png)
属实是天坑

命令执行没问题了，尝试写一句话，
果不其然被拦截了

尝试加点注释

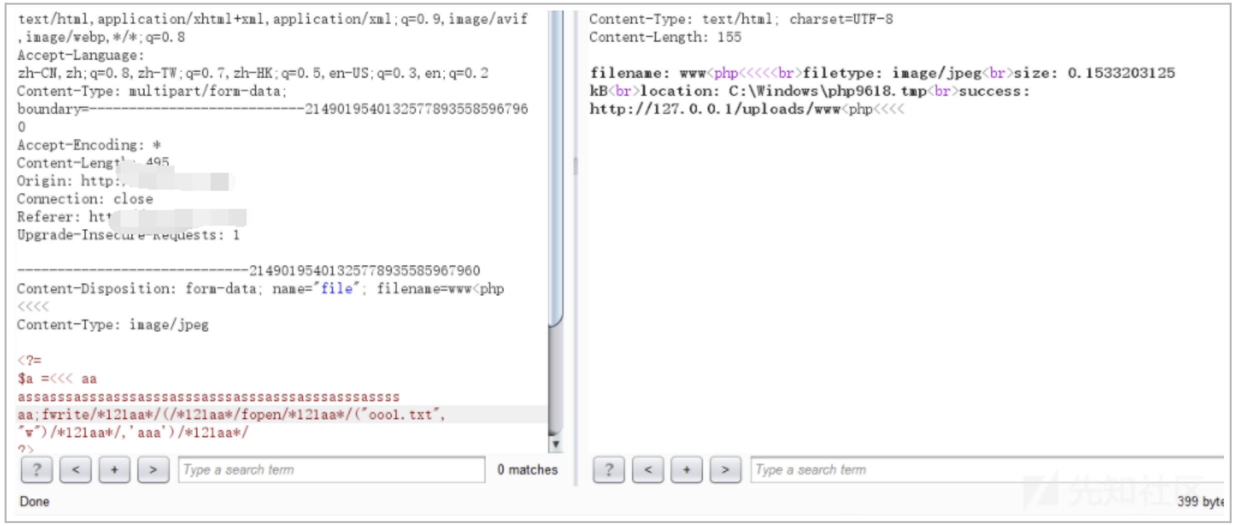
(https://xzfile.aliyuncs.com/media/upload/picture/20211101102149-784e3f4e-3aba-1.png)
没问题过去了，访问一下

(https://xzfile.aliyuncs.com/media/upload/picture/20211101102209-84354abe-3aba-1.png)
又转起来了

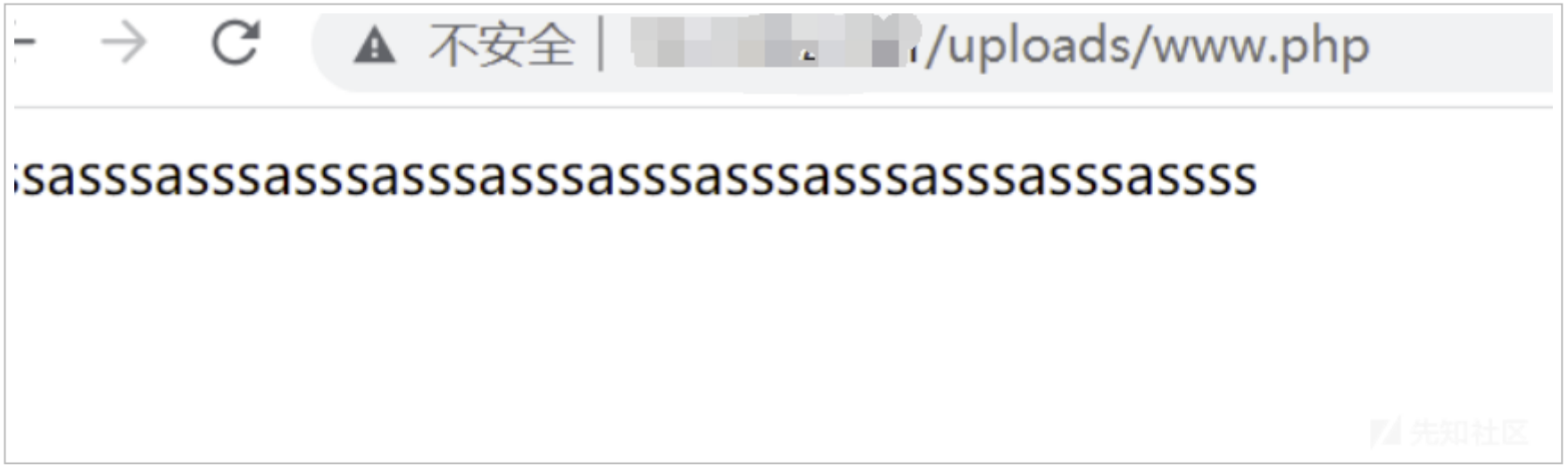
本地试了一下，发现 eval 这句不能在 aa; 后面，又是语法错误

既然如此，那我只能放大招了，那就是 fopen 大法，也就是通过 php 脚本文件向服务器写新文件，达到绕过流量层 waf 检测。

先写个 txt 试试水，所有字符能放开的最好直接用注释分开，

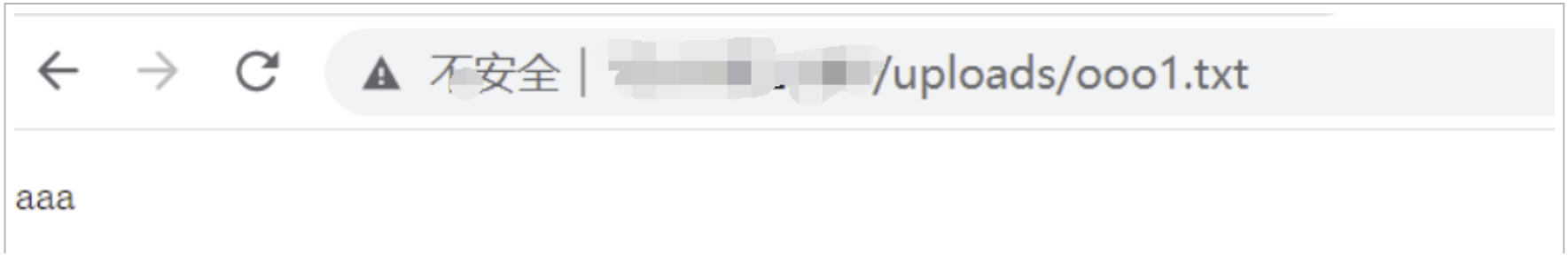


(https://xzfile.aliyuncs.com/media/upload/picture/20211101102234-92d57526-3aba-1.png)
访问一下 www.php



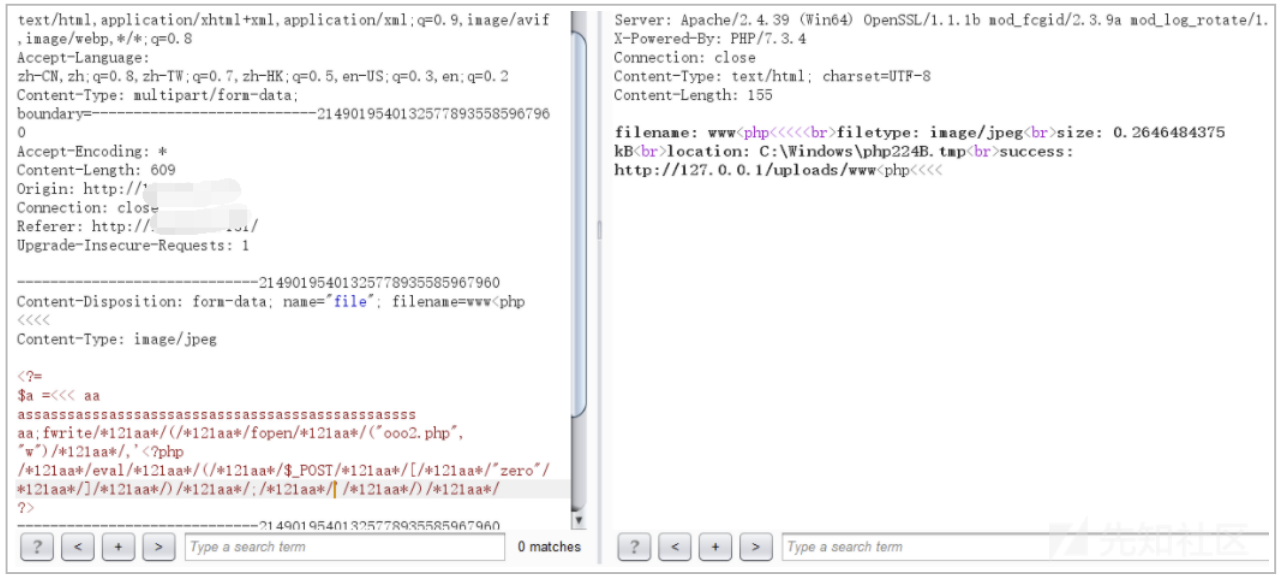
(https://xzfile.aliyuncs.com/media/upload/picture/20211101102249-9bddec02-3aba-1.png)

没问题，说明确实执行了我们的代码。

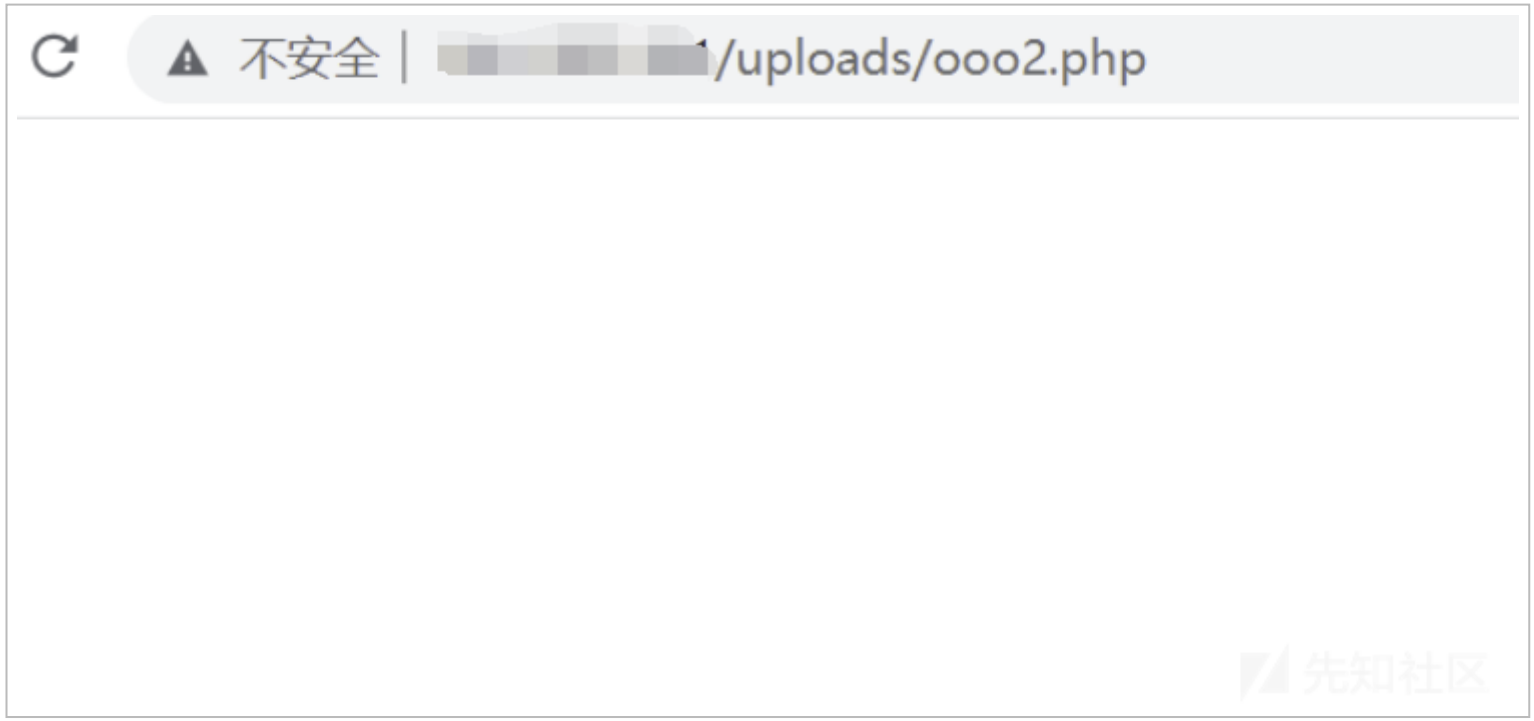


(https://xzfile.aliyuncs.com/media/upload/picture/20211101102308-a760c2ca-3aba-1.png)

那既然如此就好办多了，直接传个一句话



(https://xzfile.aliyuncs.com/media/upload/picture/20211101102328-b303f2be-3aba-1.png)



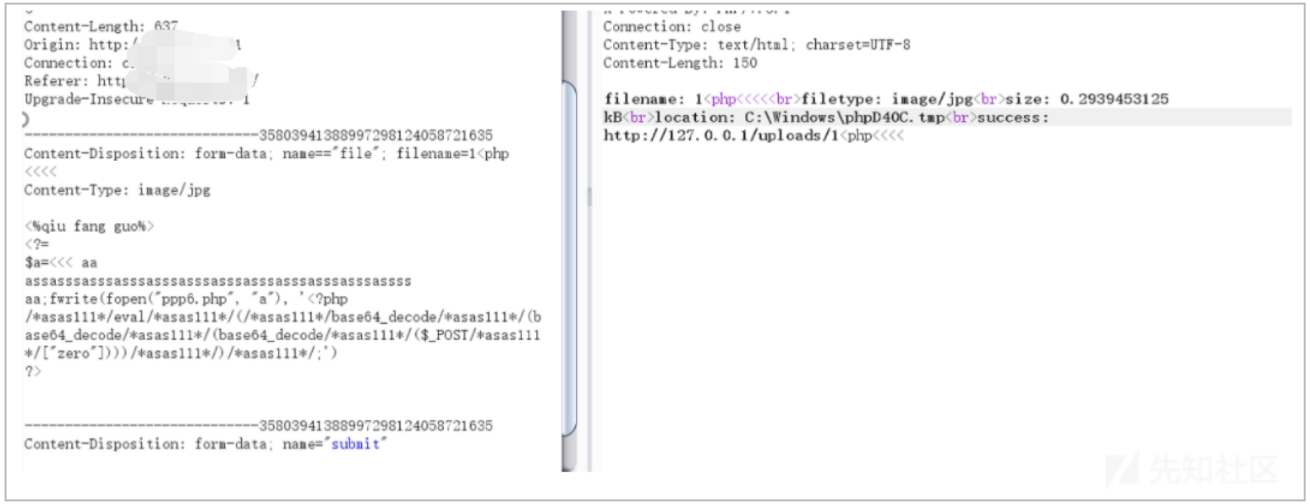
(https://xzfile.aliyuncs.com/media/upload/picture/20211101102339-b9c82cdc-3aba-1.png)

可惜，明文传输直接挂，流量还得过。

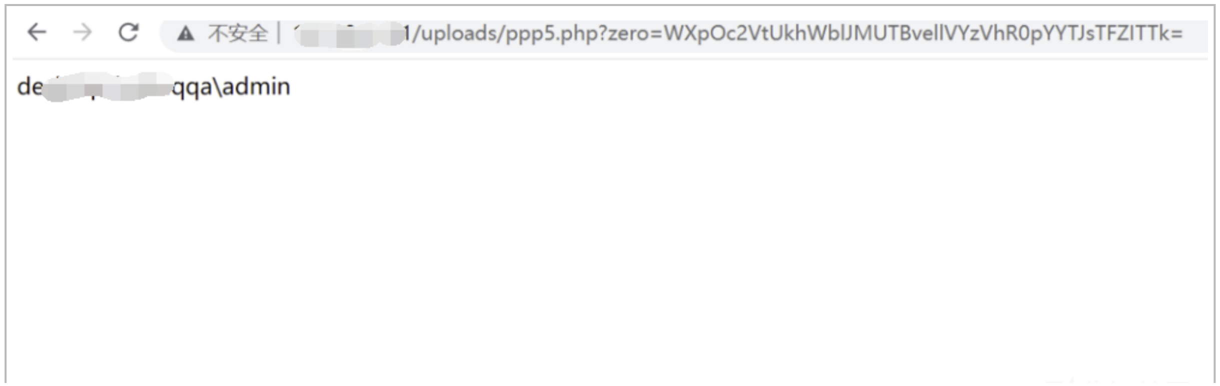
0x03：流量绕过

流量绕过就比较简单了，多次编码即可，这里上传了一个三次 base64 解密的马

流量绕过就比较简单了，多次编码即可，这里上传了一个三次 base64 解码的马



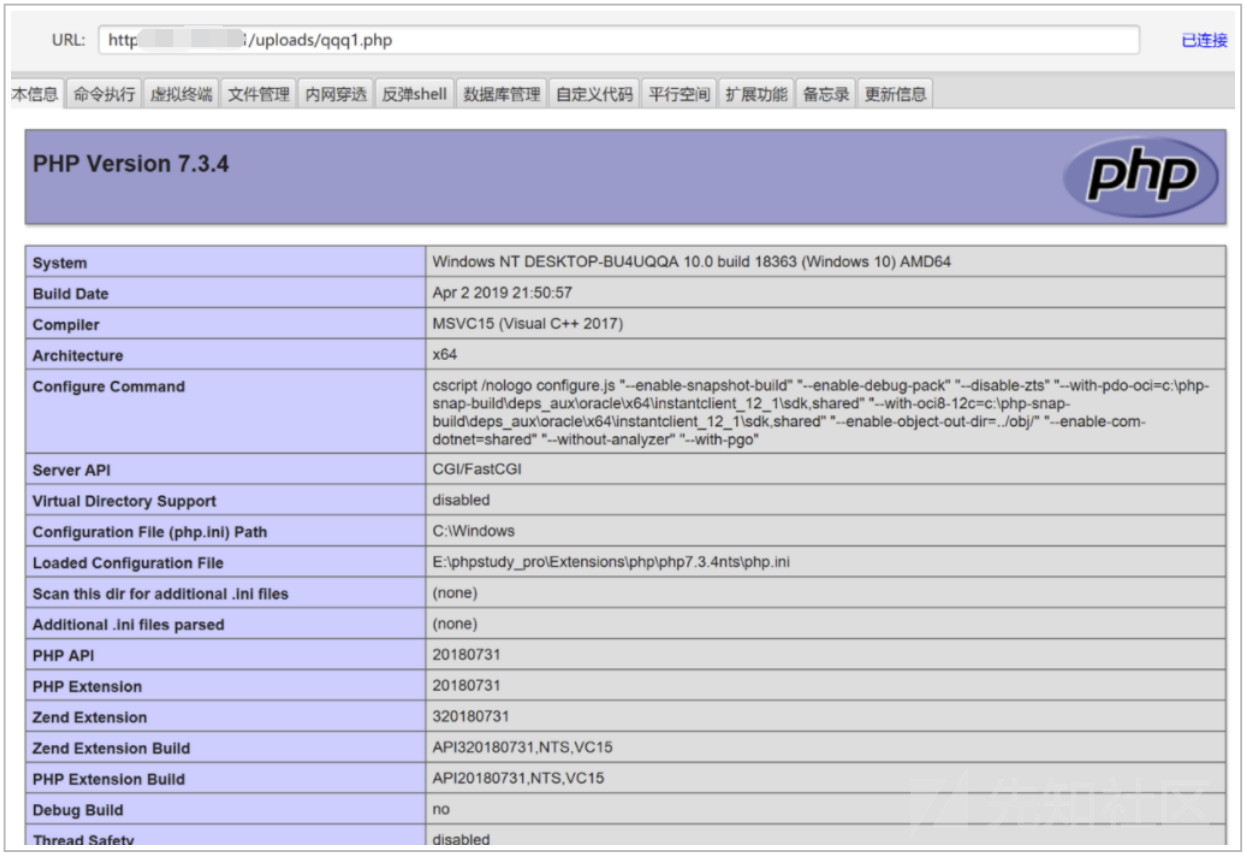
(https://xzfile.aliyuncs.com/media/upload/picture/20211101102414-ce7c05d6-3aba-1.png)



(https://xzfile.aliyuncs.com/media/upload/picture/20211101102430-d7d605fa-3aba-1.png)

但是蚁剑编码器出现了问题，可能是多次编码导致不知道哪里出了问题，直接上传冰蝎（这里需要把冰蝎分两段传，用 fopen 的 a 参数拼接脚本）

over



(https://xzfile.aliyuncs.com/media/upload/picture/20211101102451-e4c06198-3aba-1.png)