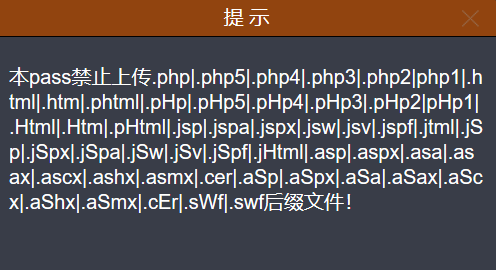
Pass04

打开第四关先查看提示

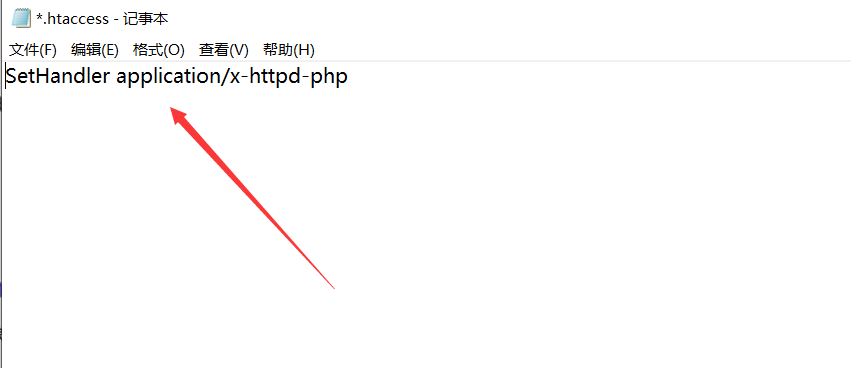


可以看到这一关的过滤太狠了，基本把该过滤的都多虑了，把大小写绕过和上面一关的配置漏洞都考虑进去了。

但是黑名单里面还是漏掉了一个文件名，那就是Apache服务器的一个配置文件.htaccess。有关这个文件的一些知识请看本站的另一篇文章《.htaccess 文件 在文件上传漏洞中的作用（附phpstudy2019复现失败详情）》

所以这一关可以先上传一个.htaccess配置文件，写入规则让服务器把指定的文件格式按照php的格式解析。然后就上传指定文件格式的文件（不在黑名单中的）进行测试。

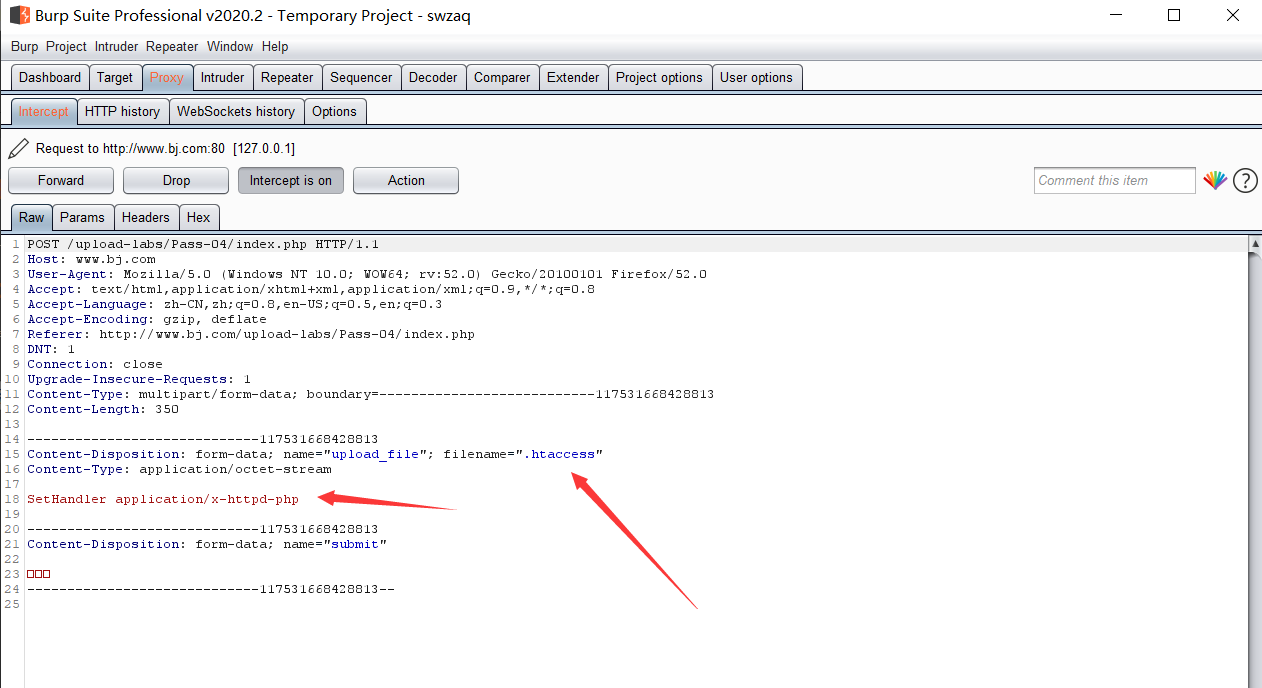
第一步：创建.htaccess文件

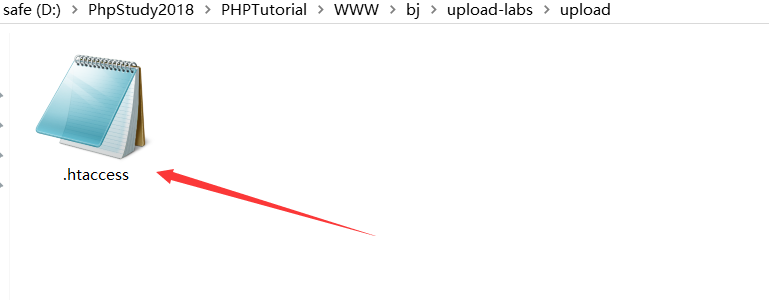


SetHandler application/x-httpd-php

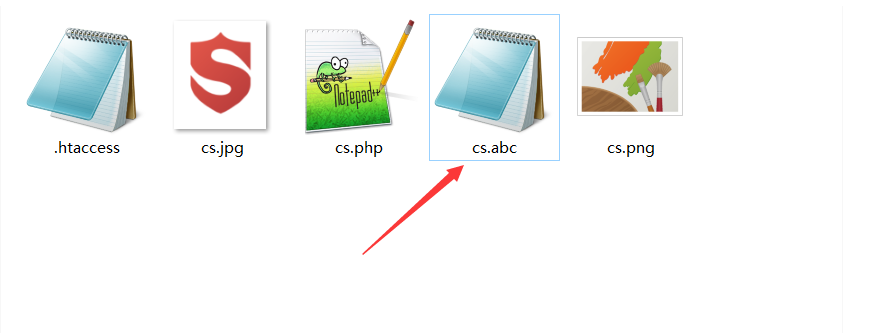
该配置的意思就是让Apache对当前目录中的所有文件都以php的格式进行解析。

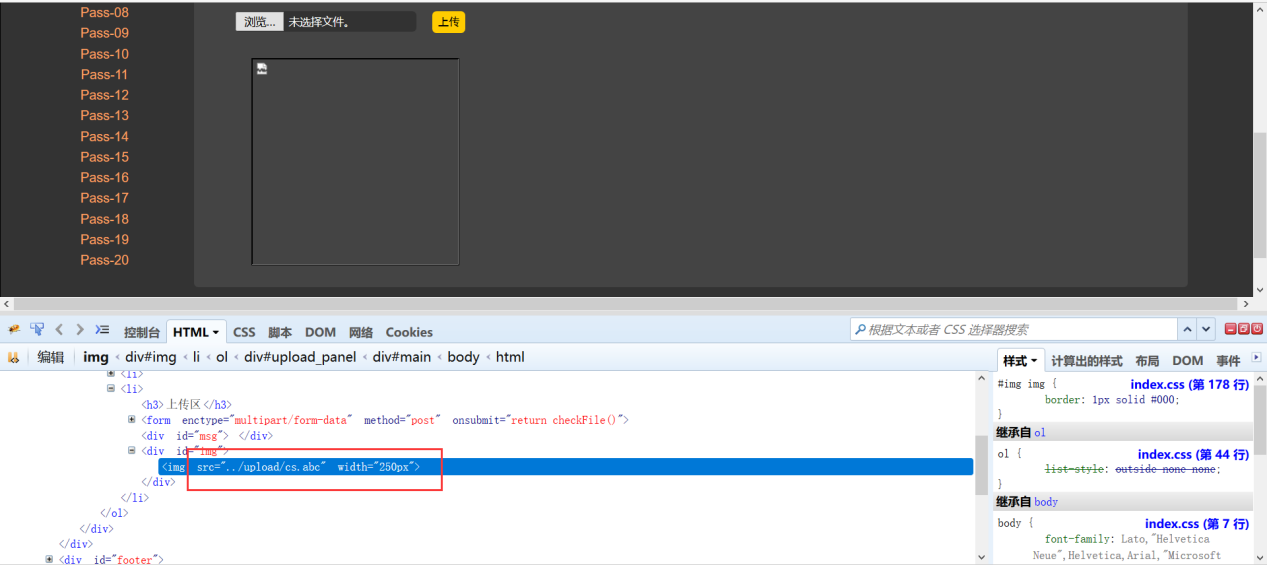
第二步：上传.htaccess文件



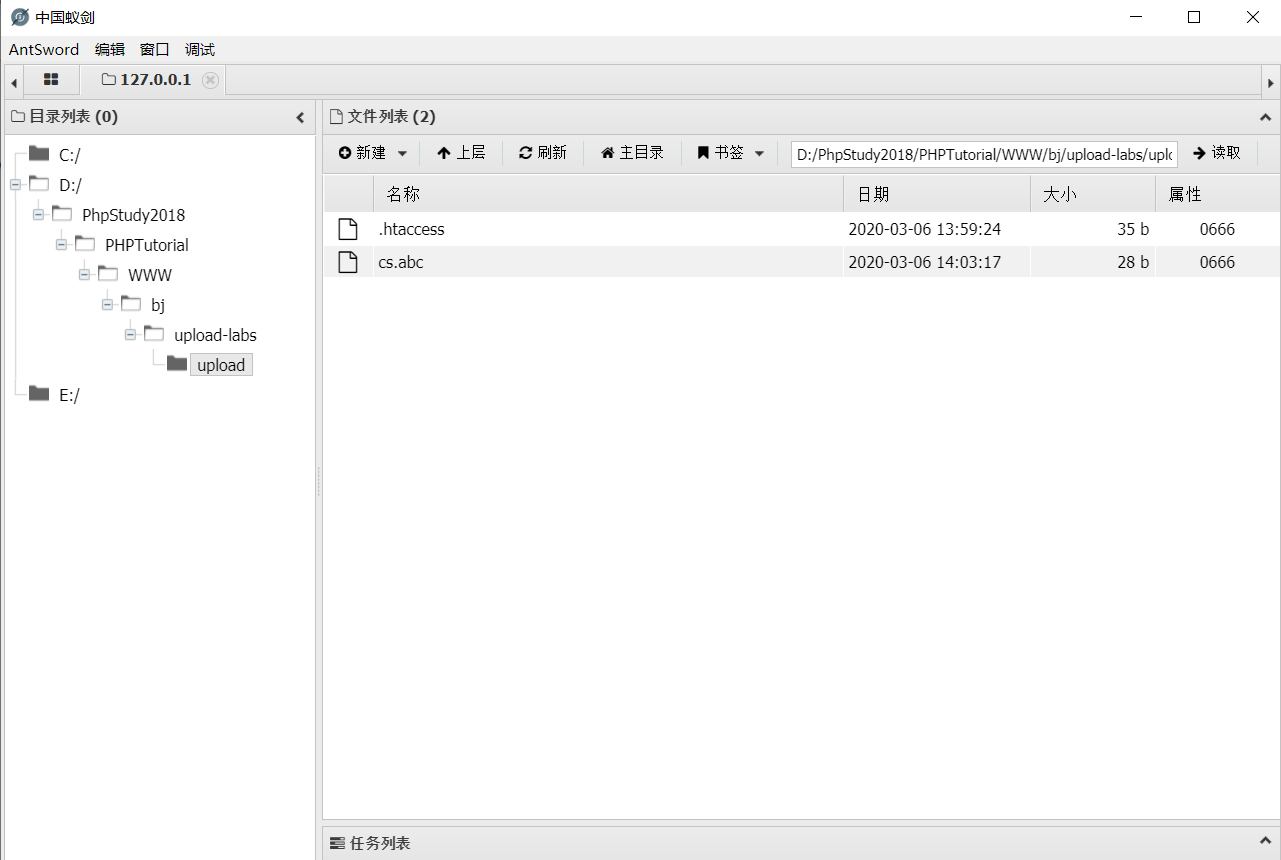


第三步：把一句话shell的后缀名随便改一下，只要不是黑名单禁止的就行



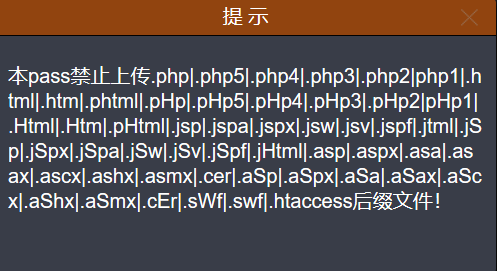


用蚁剑连接测试一下，发现确实是解析了



Pass05

打开第五关首先查看提示



哎呀我去，居然把.htaccess这个漏洞也补上了。难搞额，只能好好看看源码了。

$is\_upload = false;$msg = null;if (isset($\_POST['submit'])) {

if (file\_exists(UPLOAD\_PATH)) {

$deny\_ext = array(".php",".php5",".php4",".php3",".php2",".html",".htm",".phtml",".pht",".pHp",".pHp5",".pHp4",".pHp3",".pHp2",".Html",".Htm",".pHtml",".jsp",".jspa",".jspx",".jsw",".jsv",".jspf",".jtml",".jSp",".jSpx",".jSpa",".jSw",".jSv",".jSpf",".jHtml",".asp",".aspx",".asa",".asax",".ascx",".ashx",".asmx",".cer",".aSp",".aSpx",".aSa",".aSax",".aScx",".aShx",".aSmx",".cEr",".sWf",".swf",".htaccess");

$file\_name = trim($\_FILES['upload\_file']['name']);

$file\_name = deldot($file\_name);//删除文件名末尾的点

$file\_ext = strrchr($file\_name, '.');

$file\_ext = str\_ireplace('::$DATA', '', $file\_ext);//去除字符串::$DATA

$file\_ext = trim($file\_ext); //首尾去空

if (!in\_array($file\_ext, $deny\_ext)) {

$temp\_file = $\_FILES['upload\_file']['tmp\_name'];

$img\_path = UPLOAD\_PATH.'/'.date("YmdHis").rand(1000,9999).$file\_ext;

if (move\_uploaded\_file($temp\_file, $img\_path)) {

$is\_upload = true;

} else {

$msg = '上传出错！';

}

} else {

$msg = '此文件类型不允许上传！';

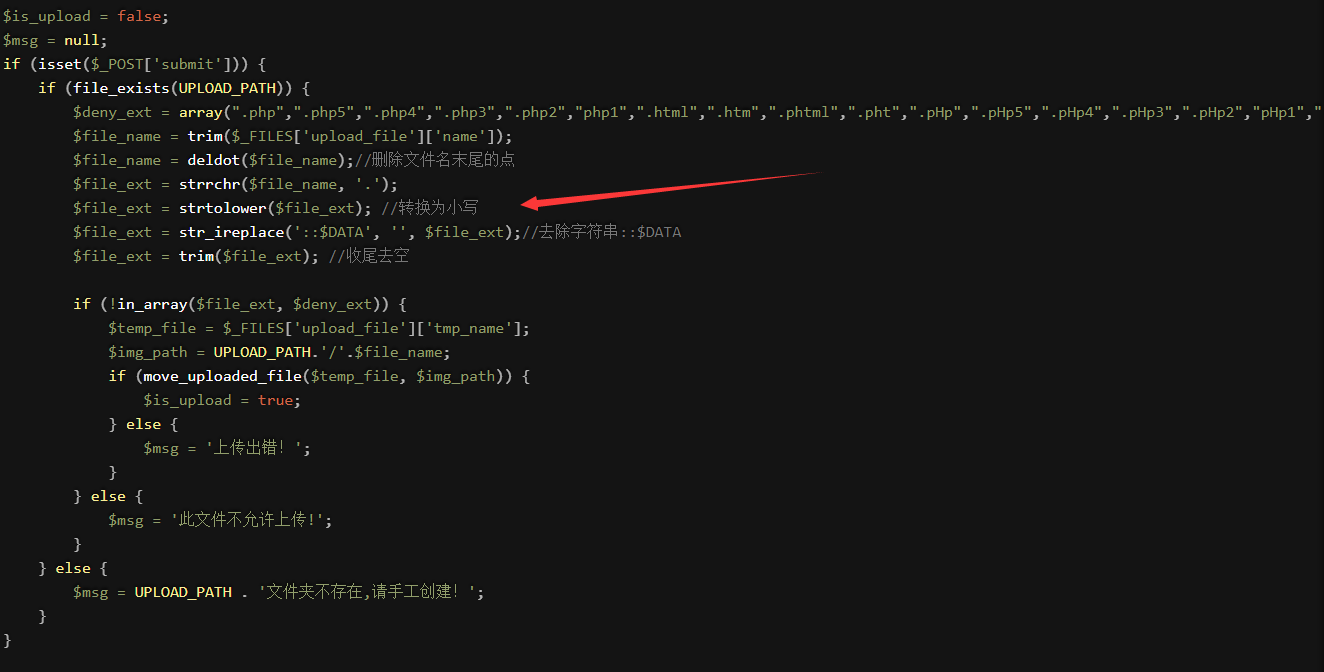
}

} else {

$msg = UPLOAD\_PATH . '文件夹不存在,请手工创建！';

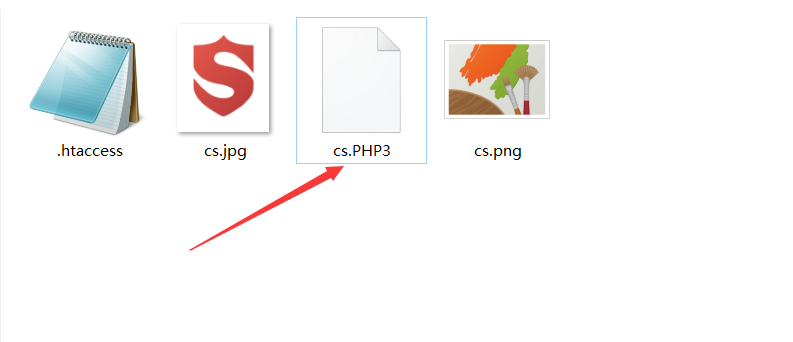
}}

同第四关对比，通过源码可以知道本关对文件后缀名没有进行大小写转换

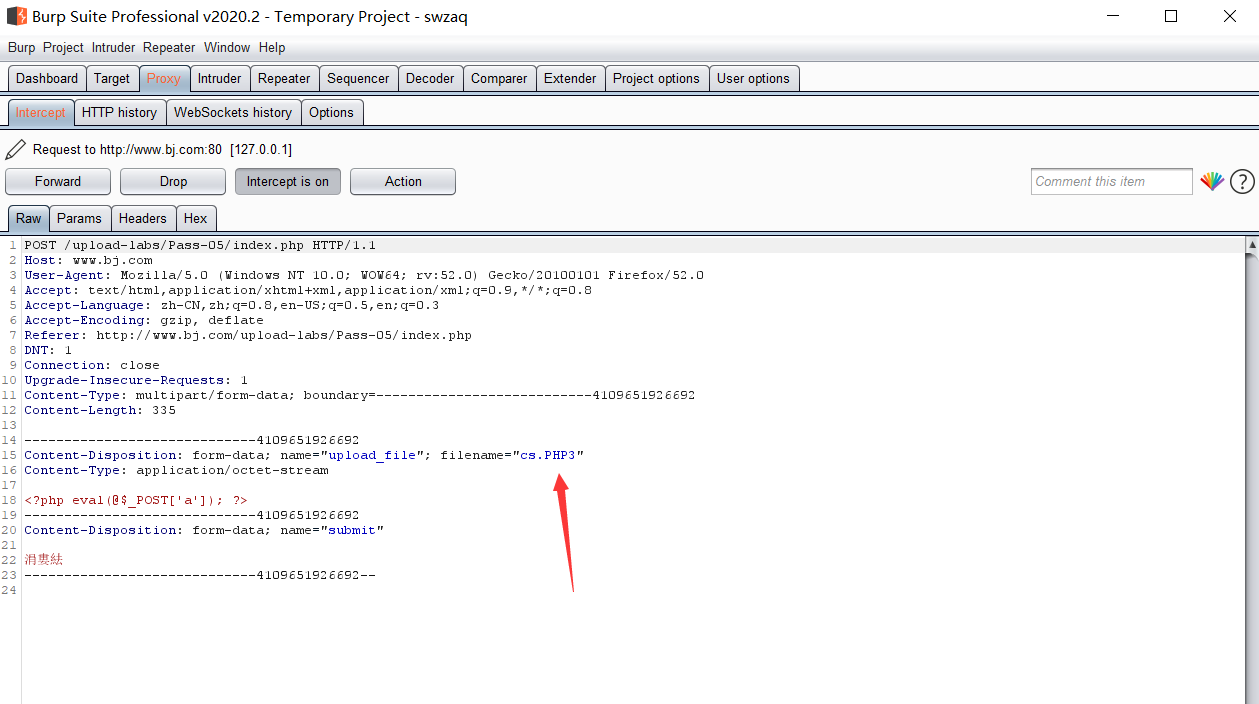


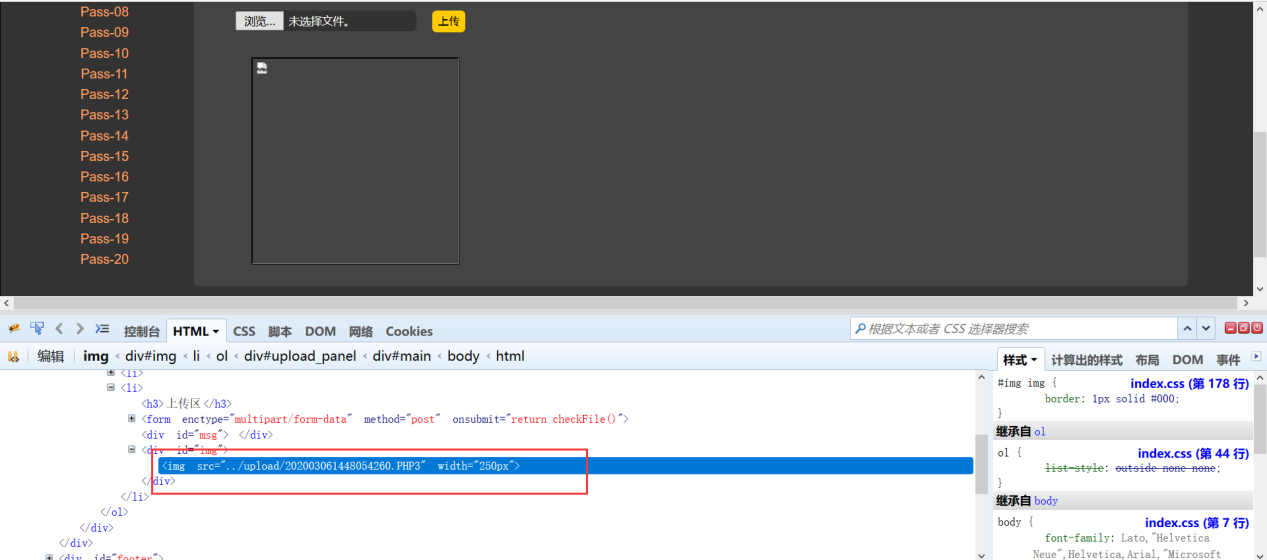
如果不将后缀名转换为小写的话，完全就可以写成PHP3之类的绕过黑名单。因为黑名单只是禁止了php3和pHp3，当有小写转换的时候，无论怎么写都绕不过php3这个禁止，现在就不行了。

第一步：将一句话shell后缀名改为PHP3



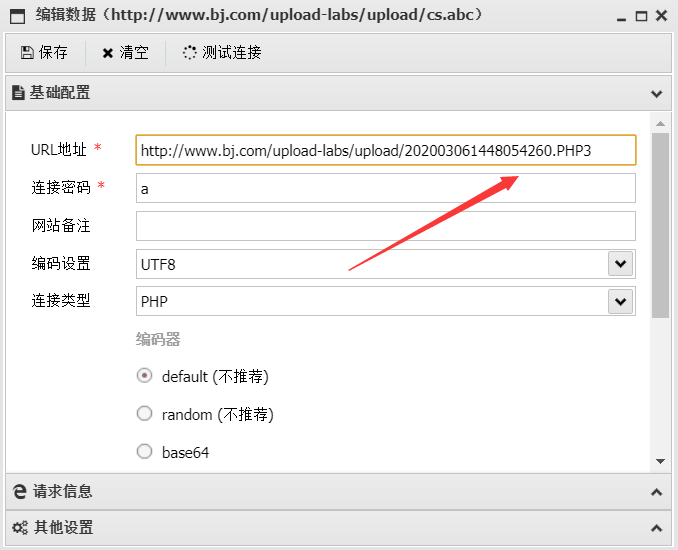
第二步：上传文件





发现确实上传成功，只不过文件名被修改了。

我们用蚁剑连接测试



最终成功连接上了，说明确实绕过了黑名单并且服务器也对文件进行了解析。