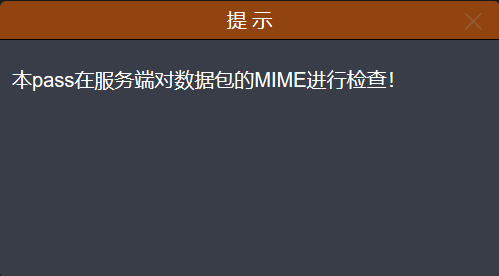
Pass02

打开靶场还是为了上传一个shell，这次首先查看提示



原来这一关考的不再是js前端校验了，而是服务器端的MIME校验了。那么先了解一下MIME是什么吧。



对于web应用来说的话，就是在把响应结果传送到浏览器上的时候，浏览器必须启动适当的应用程序来处理这个响应文档。这就可以通过多种类型MIME（多功能网际邮件扩充协议）来完成。在HTTP中，MIME类型被定义在Content-Type header中。

比如你要传送一个Microsoft Excel文件到客户端。那么这时的MIME类型就是“application/vnd.ms-excel”。 在大多数实际情况中，这个文件会被传送给 Execl来处理（假设我们设定Execl为处理特殊MIME类型的应用程序）。

在HTTP协议中，MIME类型就被定义在数据包的**Content-Type** 字段中了。那么MIME跟上传又有什么关系呐？让我们来看看源码

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

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

if (($\_FILES['upload\_file']['type'] == 'image/jpeg') || ($\_FILES['upload\_file']['type'] == 'image/png') || ($\_FILES['upload\_file']['type'] == 'image/gif')) {

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

$img\_path = UPLOAD\_PATH . '/' . $\_FILES['upload\_file']['name']

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

$is\_upload = true;

} else {

$msg = '上传出错！';

}

} else {

$msg = '文件类型不正确，请重新上传！';

}

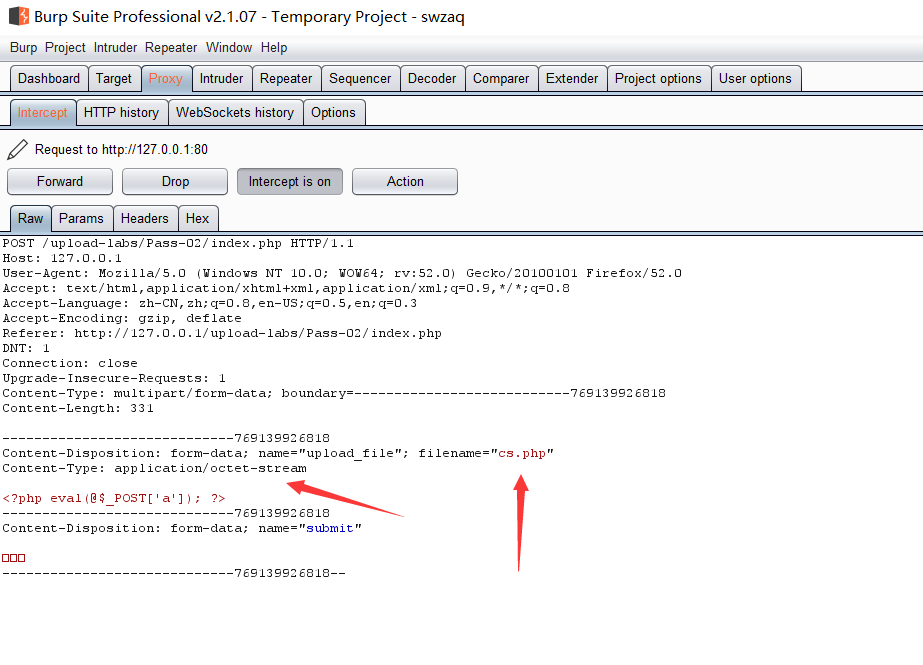
} else {

$msg = UPLOAD\_PATH.'文件夹不存在,请手工创建！';

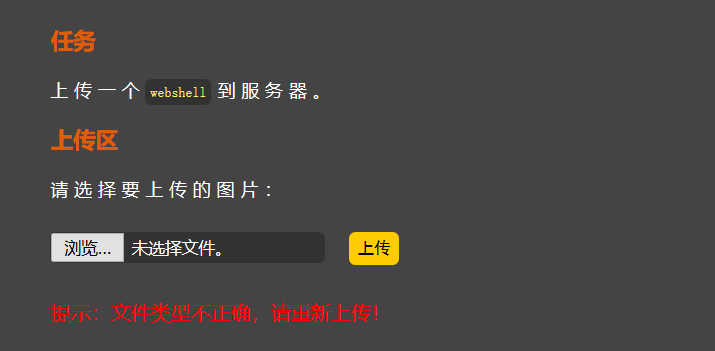
}}

从源码可以看出，在这一关中，服务器端对上传到服务器的文件进行了类型检测，也就是检测MIME。如果类型是image/jpeg、image/png、image/gif中的任何一种就允许上传，否则就提示文件类型不正确。

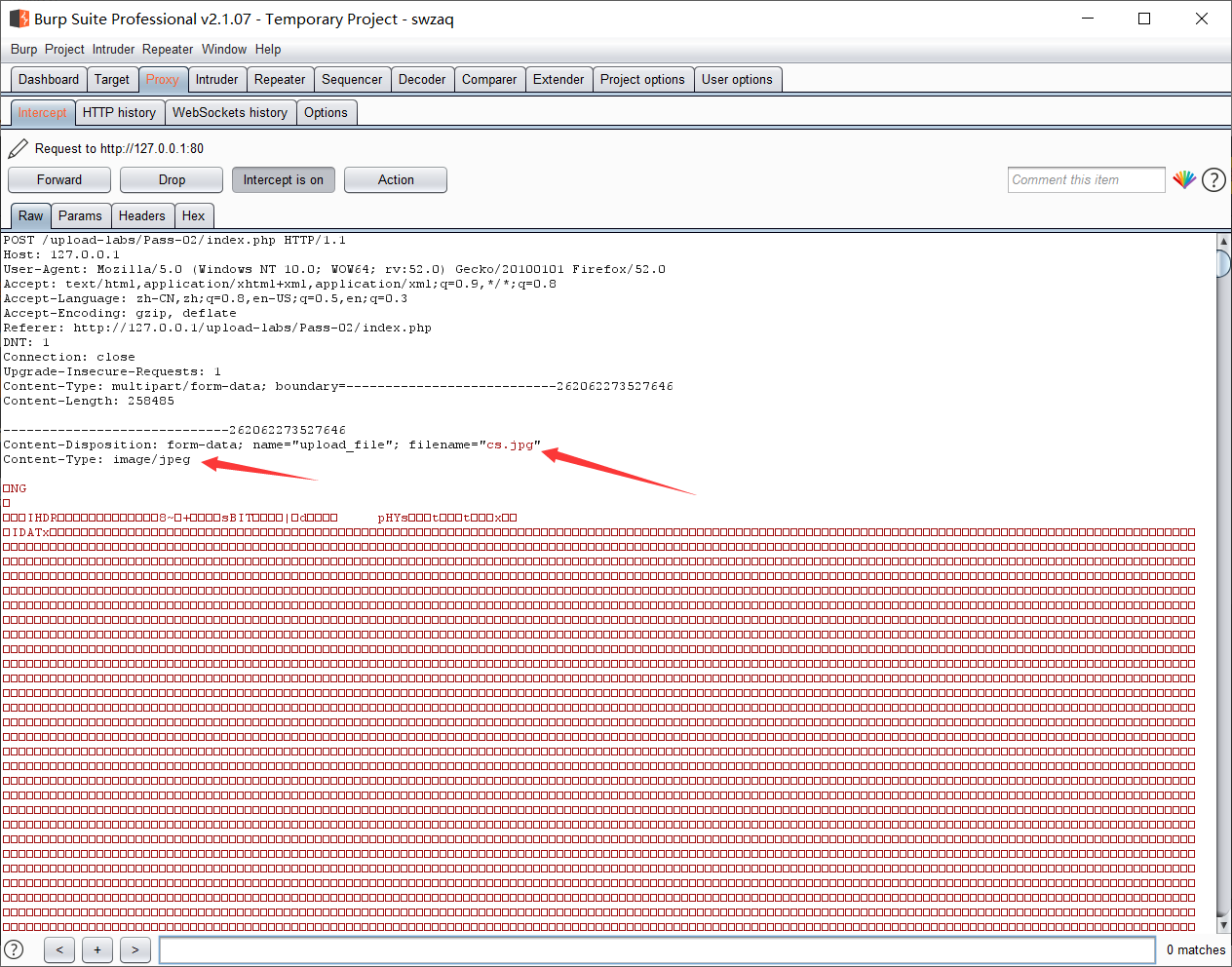
那么我们先直接上传一个php的看看。（记得抓包测试）



可以看到上传php文件的时候，MIME类型明显是不符合服务器端的要求的，那么放包看看能不能上传成功。



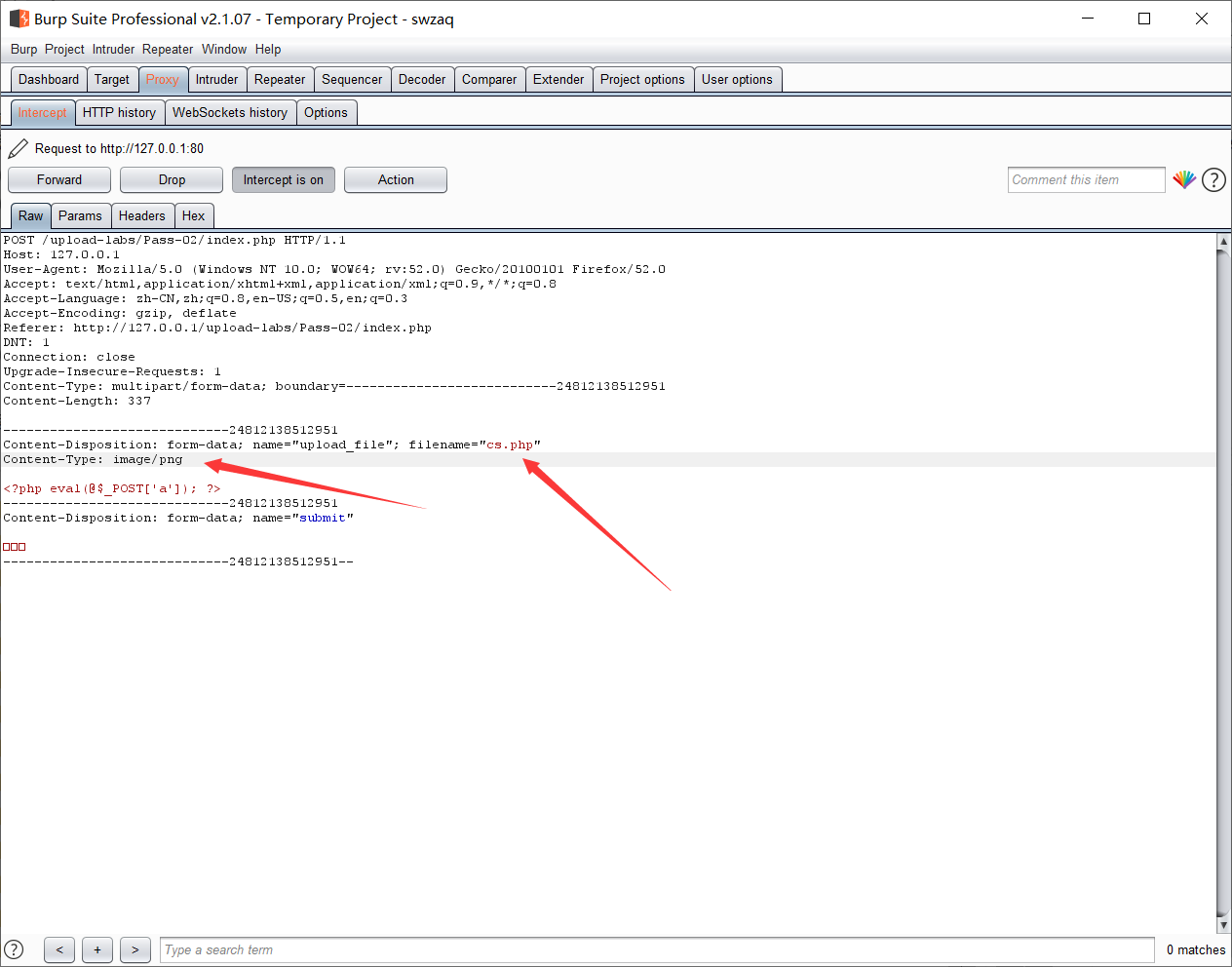
果然提示了类型不正确。那么再上传一个正常图片看看





看来只要是MIME类型满足了之后确实就上传成功了。那么如何才能将php文件成功上传到服务器呐？

这里就需要在上传shell的时候把php文件的MIME类型改为服务器允许的就可以。这样仅仅就是绕过了服务器端对MIME的检测并成功将shell上传，并不会对php文件本身造成什么改动。

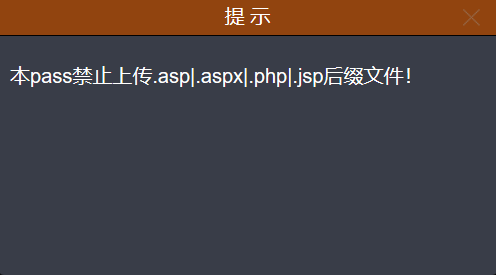




果然修改之后绕过了服务器端的检测。

Pass03

打开第三关查看提示



这下好了，直接把所有的常用脚本语言的shell文件后缀全部禁止了。那么这样就万无一失没有漏洞了吗？其实不然。

当黑名单把所有脚本后缀都禁止的时候，还会由于管理人员配置不当或者Apache本身的漏洞而造成漏洞的。

这里我们补充一点关于Apache的知识：

1. Apache解析顺序是从右到左开始解析文件后缀的，如果最右侧扩展名不能识别的话，就继续往左判断，直到遇到可以解析的文件后缀为止。
2. Apache的httpd.conf中可以进行配置，规定哪些文件后缀是以php格式来解析的，这是属于管理员的配置不当导致的漏洞。

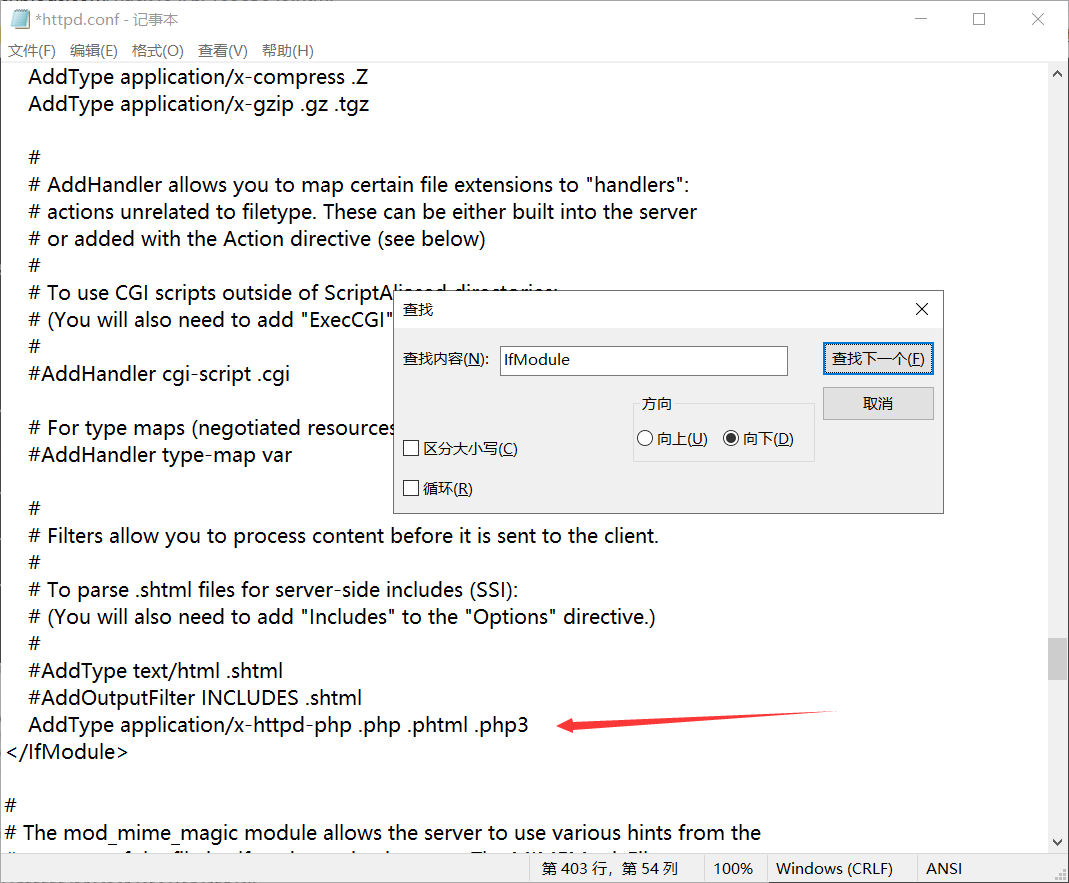
这里我们通关的方式就选择配置不当这种，因为集成环境中php是作为Apache的模块来工作的，一些解析配置比较安全，虽然Apache本身还是存在解析漏洞，但是php工作的时候则不会按照它的规则来执行（解析漏洞会专门写一篇文章详细说明）。

要用第二种方式通关的话，需要对httpd.conf文件进行修改。

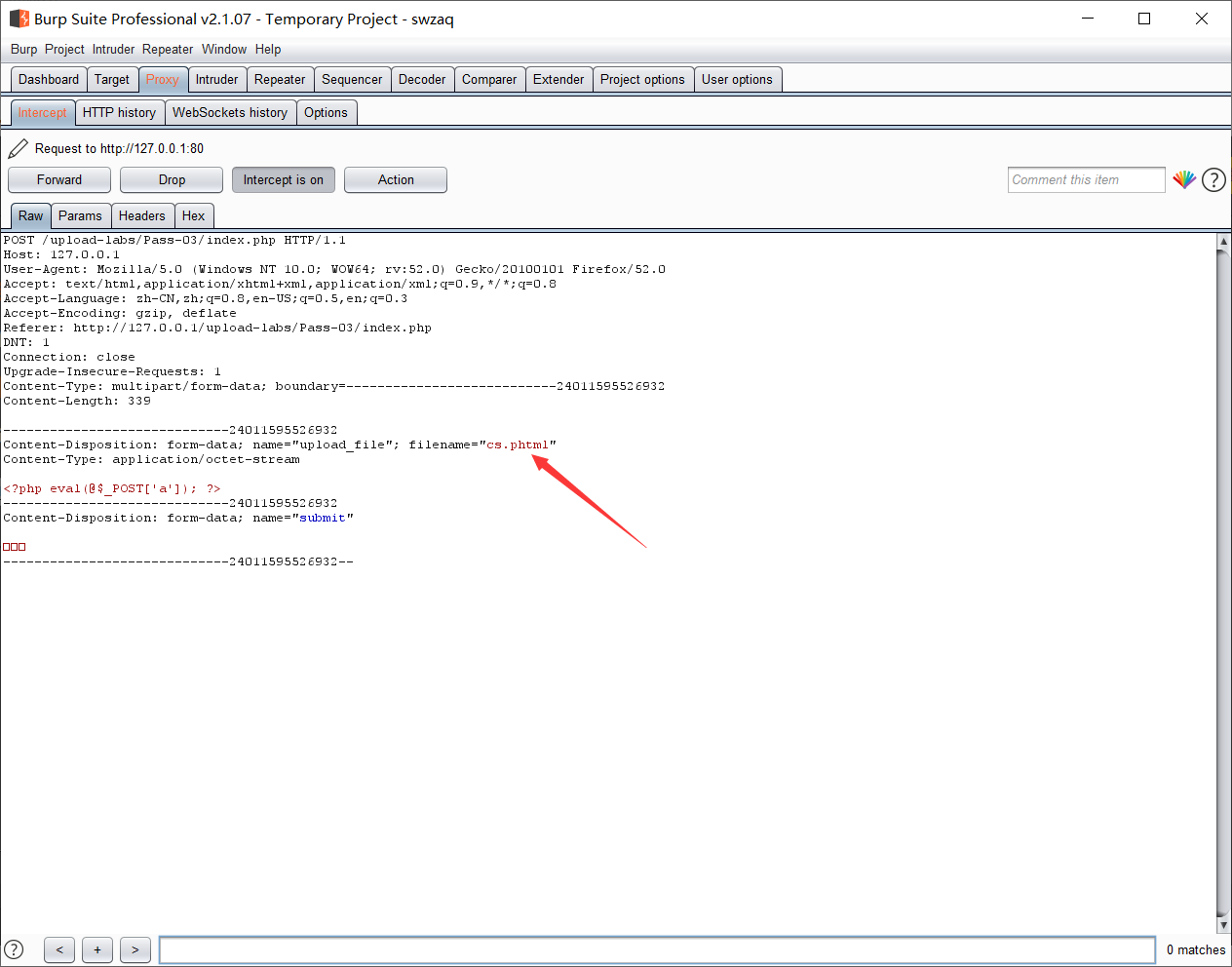
要有形同下面这样的配置

AddType application/x-httpd-php .php .phtml .php3

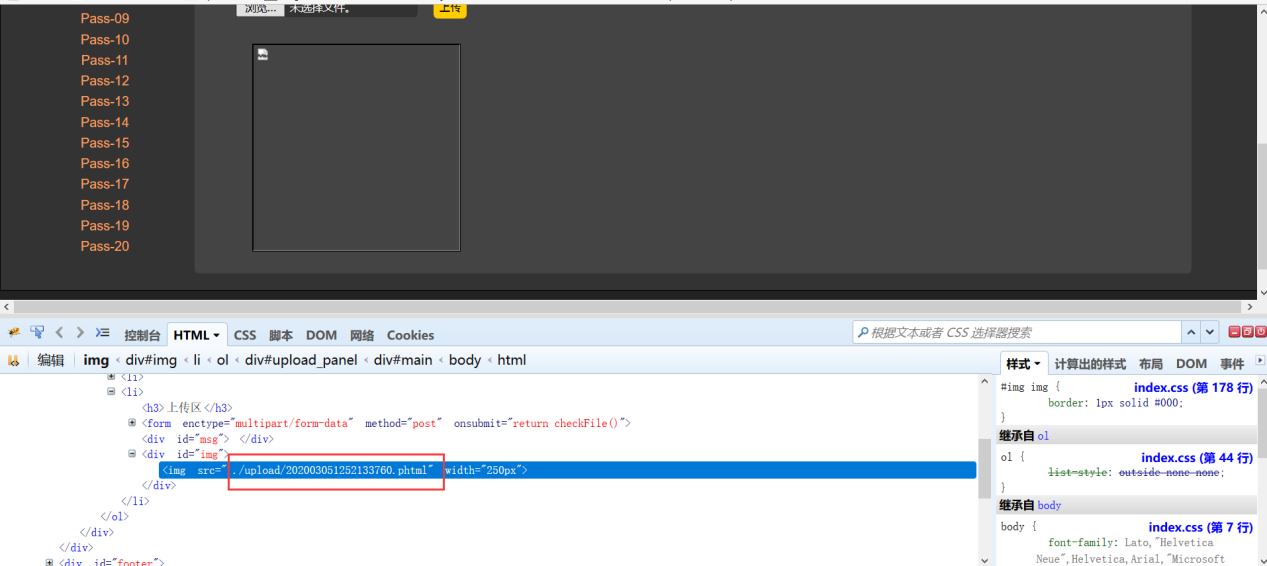
这句话的意思就是服务器会把.php、.phtml、.php3这几个后缀名的文件都用php格式来解析。



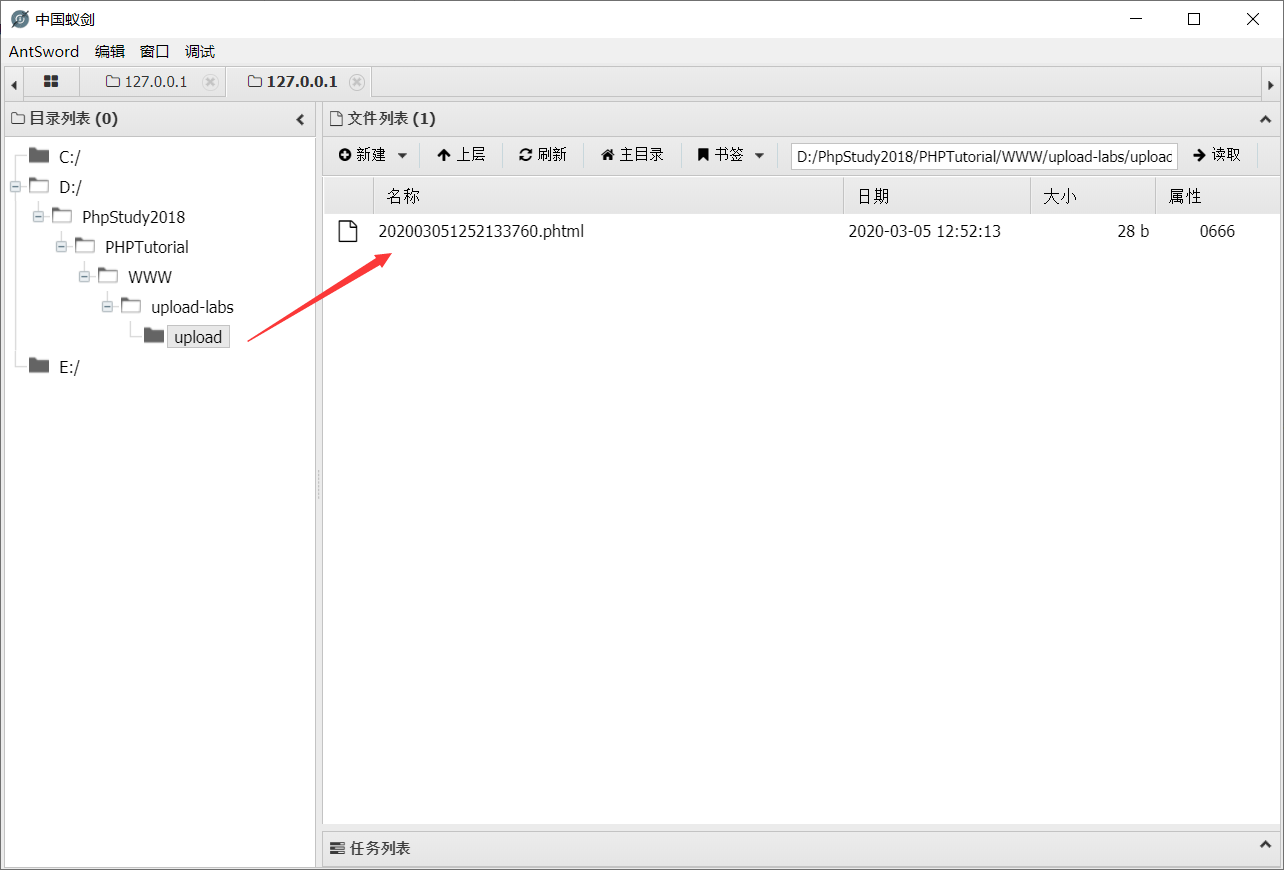
修改保存之后重启服务器进行测试



通过这种方式上传的文件服务器会修改文件名



上传成功之后，我们用蚁剑链接试试



成功绕过服务器端黑名单getshell。