**前言**

Swagger是一个规范和完整的框架，用于生成、描述、调用和可视化 RESTful 风格的 Web 服务。总体目标是使客户端和文件系统作为服务器以同样的速度来更新。相关的方法，参数和模型紧密集成到服务器端的代码，允许API来始终保持同步。Swagger-UI会根据开发人员在代码中的设置来自动生成API说明文档，若存在相关的配置缺陷，攻击者可以未授权翻查Swagger接口文档，得到系统功能API接口的详细参数，再构造参数发包，通过回显获取系统大量的敏感信息。

**"绕过"**

在挖掘src的时候，我们经常会使用burp插件发现一个Spring的路由，其中最终要的莫过于swagger文档。



但有时候总是事与愿违，swagger的路径经常会提示权限不足或者文档已被删除。这时候我们怎么继续利用呢？网上常见的利用xff、..;/等等来进行权限绕过。

但是我们想一下，获取swagger文档的目的即获取一些api路径，呢么有么有其他方式来获取这些路径呢？

在某次项目中，我遇到了swagger路径访问不成功，此时我继续进行目录扫描发现存在mappings路径。



在返回的json字符串中，我发现了patterns参数竟然有很多路径。



于是记住chatpgt编写路由提取脚本

import json

# 输入的JSON数据

with open('task.txt','r') as fp:
 json\_data = fp.read()
data = json.loads(json\_data)
web\_handlers = data['contexts']['application-1']['mappings']['dispatcherServlets']['dispatcherServlet']
# 提取所有patterns
patterns = []

for handler in web\_handlers:
 try:
 patterns.extend(handler['details']['requestMappingConditions']['patterns'])
 except:
 pass

# 打印提取的patterns
print("Extracted Patterns:", patterns)



结合burp批量访问



最终发现了多个接口存在未授权，泄漏了大量敏感信息。