shiro550/721反序列化漏洞原理

一，漏洞介绍

1-1 什么是shiro

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

1-2 什么是序列化

序列化就是为了传输遍历，把一个对象类型的数据转换成字符串进行传输；或者在PHP语言里面把一个类或者对象，或者函数等通过serialize函数进行序列化便于传输；序列化后产生的JSON，或者XML格式不仅传输便利，而且可以跨语言传输数据，这个把某个对象序列化成json格式或者XML格式或者其他序列化格式的字符串过程称为序列化。

1-3 什么是反序列化

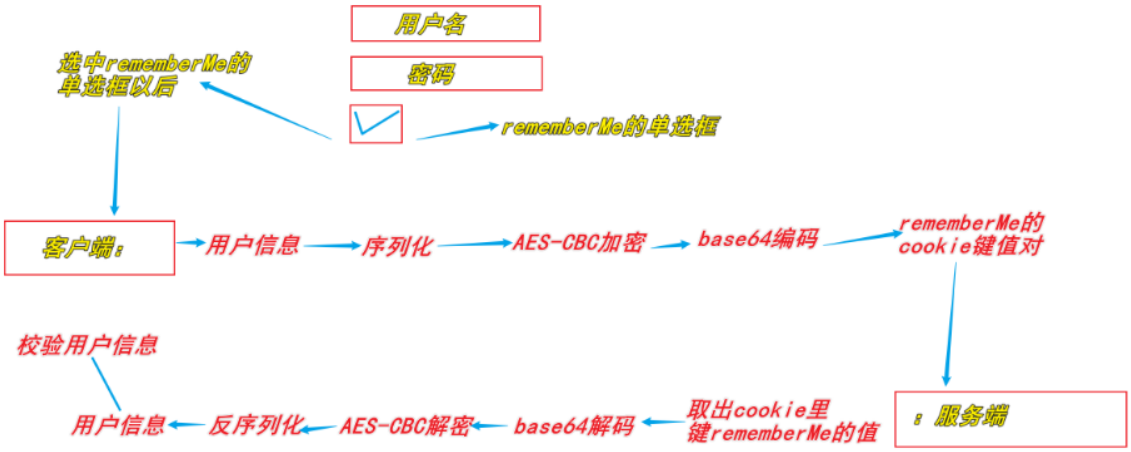
反序列化就是序列化的逆向过程，把一个序列化的JSON字符串内容或者XML内容反向还原回序列化前的对象格式

1-4 漏洞原理

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。

在Apache shiro的框架中，执行身份验证时提供了一个记住密码的功能（RememberMe），如果用户登录时勾选了这个选项。用户的请求数据包中将会在cookie字段多出一段数据，这一段数据包含了用户的身份信息，且是经过加密的。加密的过程是：用户信息=>序列化=>AES加密（这一步需要用密钥key）=>base64编码=>添加到RememberMe Cookie字段。勾选记住密码之后，下次登录时，服务端会根据客户端请求包中的cookie值进行身份验证，无需登录即可访问。那么显然，服务端进行对cookie进行验证的步骤就是：取出请求包中rememberMe的cookie值 => Base64解码=>AES解密（用到密钥key）=>反序列化。

客户端产生rememberMe键值对以及服务端进行cookie验证步骤



在服务端AES解密以后进行反序列化才得到用户信息

1-5 漏洞利用思路

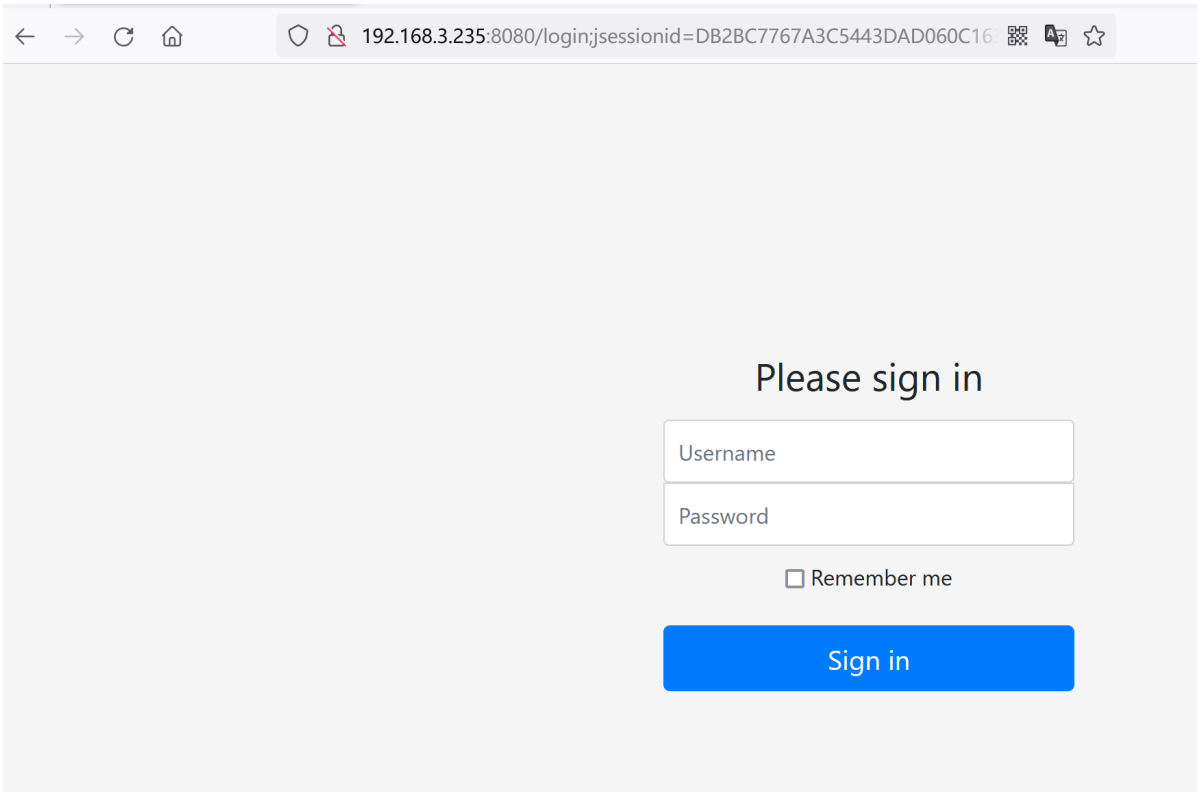
既然能进行序列化，那我们可以对我们自己的攻击代码进行相同的AES加密，base64编码以后产生rememberMe字段发给服务端，服务端反向进行解密得到我们攻击代码并会运行，进而我们就攻击成功了

二，靶场搭建

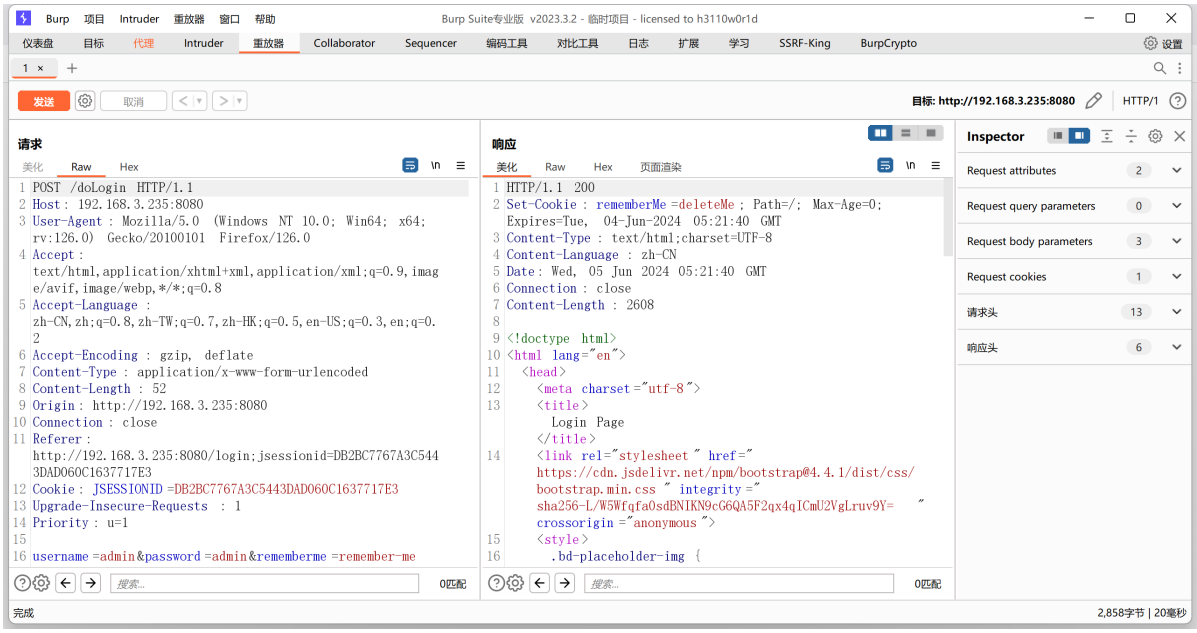
基于vulhub靶场进行搭建

```
cd vulhub-master
cd shiro
cd CVE-2016-4437
docker-compose up -d （运行靶场）
docker ps （查看容器列表）
```

浏览器访问一下看看，发现已经成功运行了



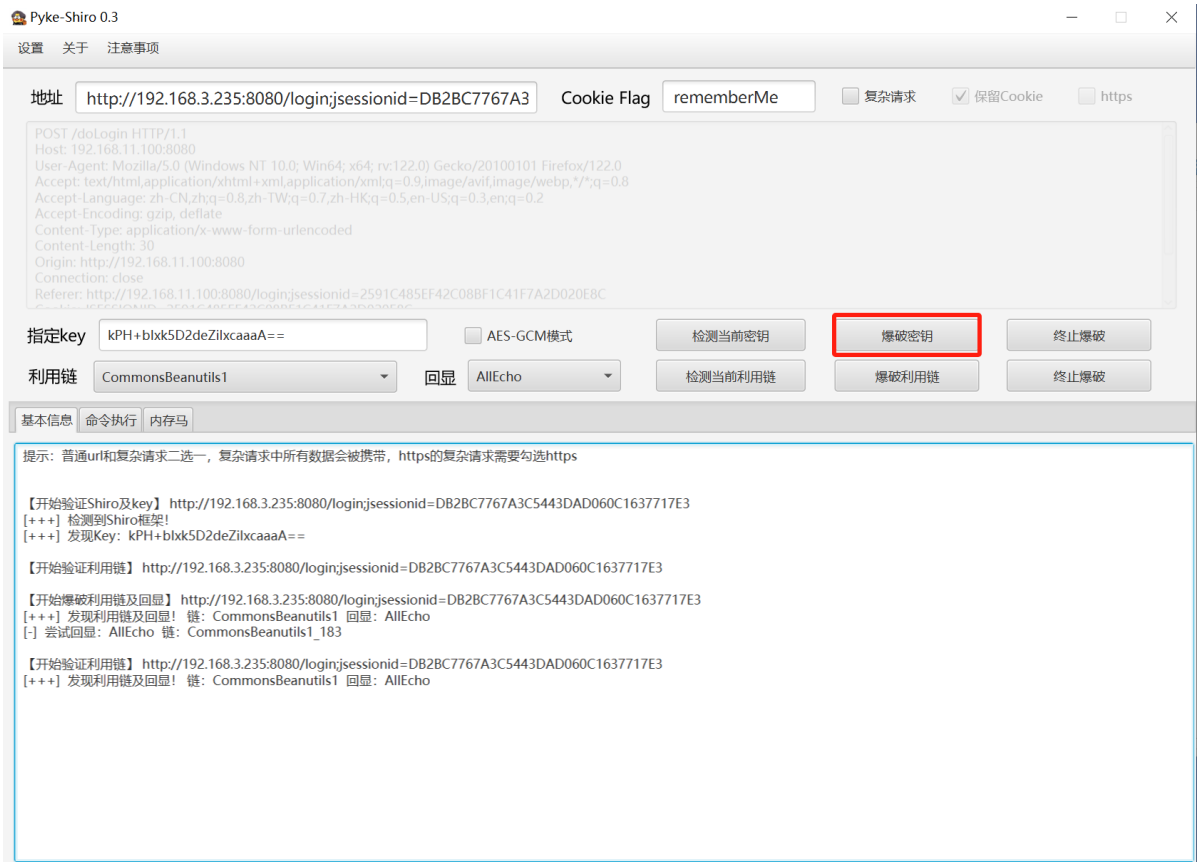
随便输入用户名和密码进行抓包，放到重放器里面，点击发送，相应包里看到rememberMe = deleteMe字段，可以说可能存在这个shiro550反序列化漏洞



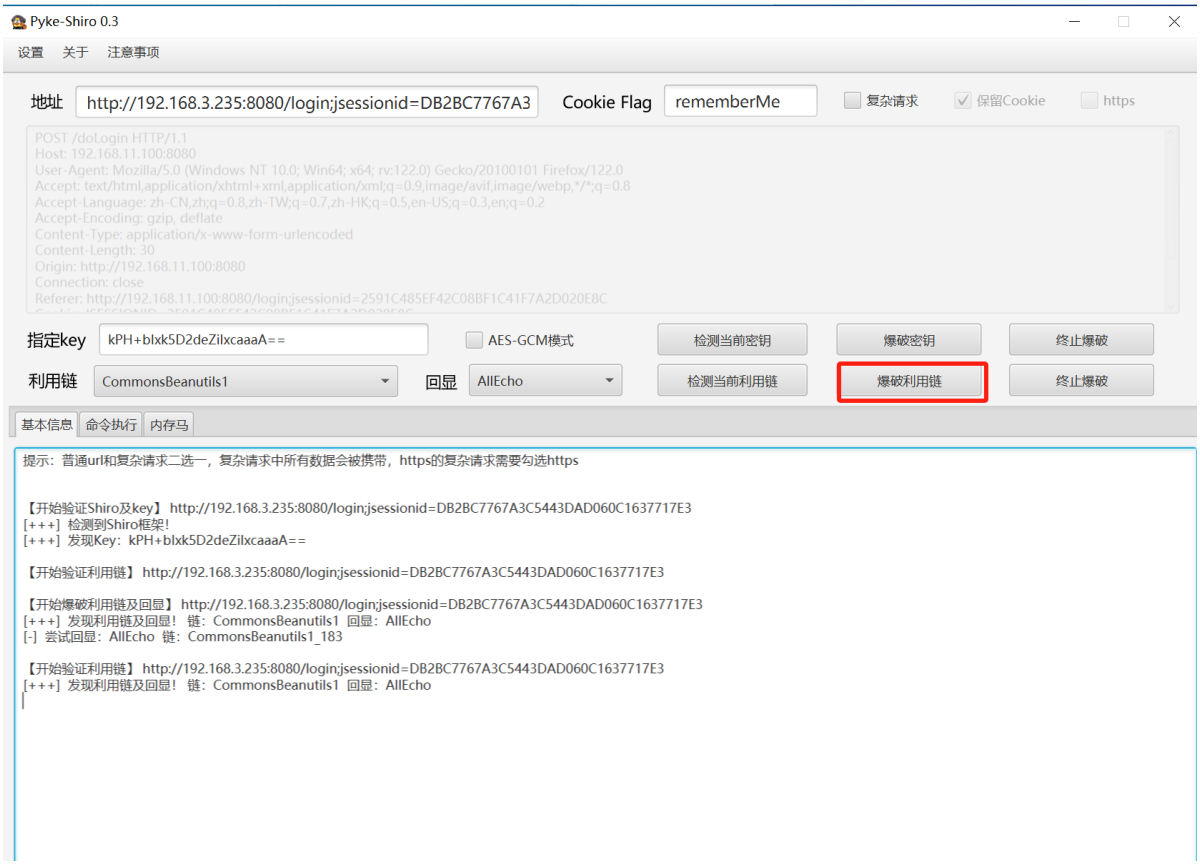
三，漏洞复现和利用

用自动化工具进行利用，下面是用工具直接进行利用

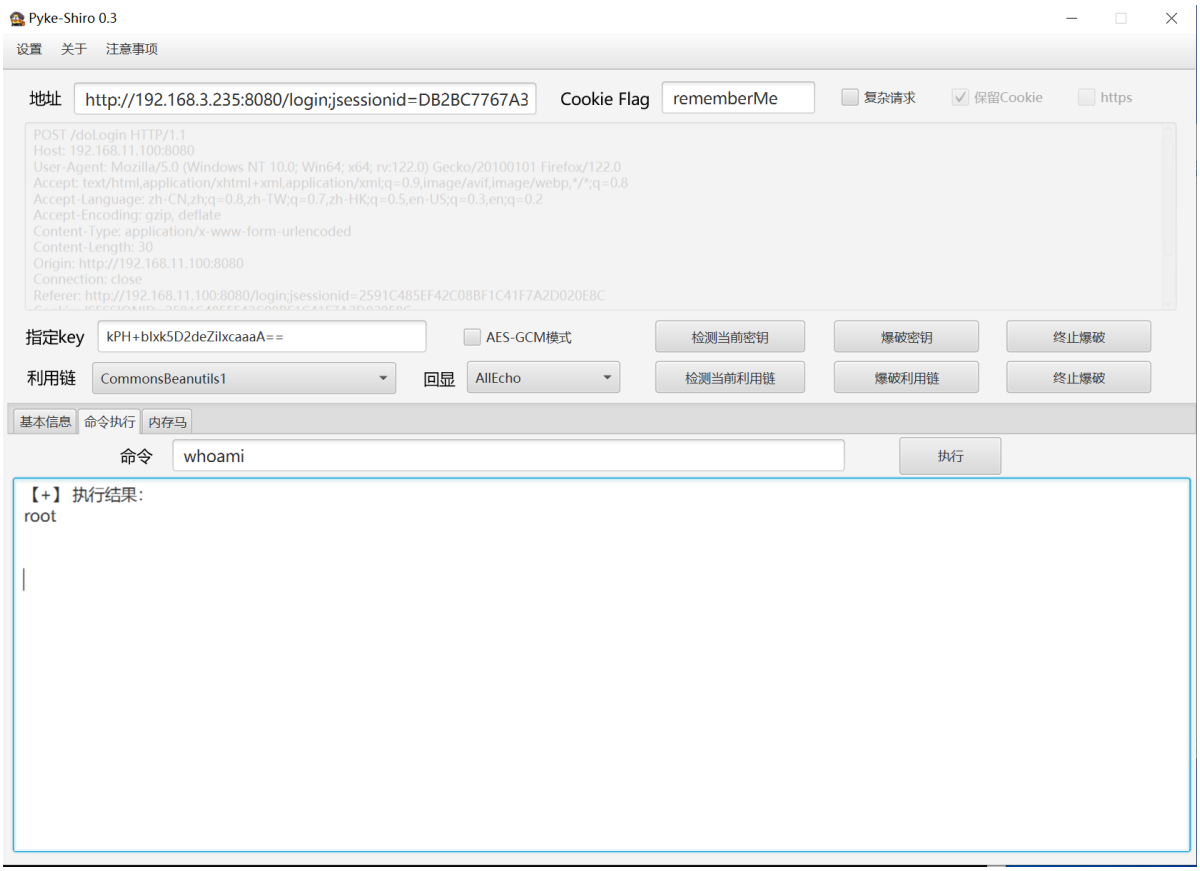
爆破密钥



根据提示选择合适的利用链以后



切换到命令执行，就可以执行系统命令了



四、修复建议

1、使用开源shiro框架时，修改默认密钥

2、代码审计，全局搜索 “setCipherKey(Base64.decode(” 关键字，或者"setCipherKey"方法，Base64.decode()中的字符串就是shiro的密钥，要确保该密钥的安全性

3、WAF拦截Cookie中长度过大的rememberMe值

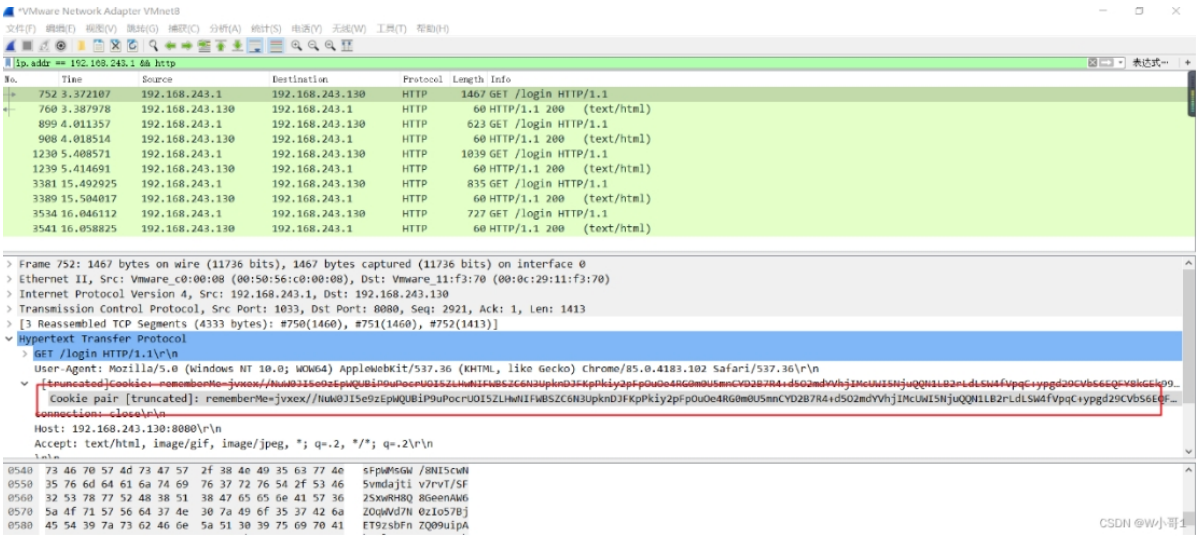
五、查看攻击特征

攻击数据包特征

例如：
rememberMe=nfSXAUiVrVcQIfpJraUD8MLp4CIVNDz/QldxOVttSOmhRlzKHTyzTVP2UxbSUU5f/NnognoRFxvGNFlWywH85c8Van8+O3Eb54iZns7+H/q/030ZgKuEu9ZMO8SghBzYZ70IZaNCjo4c1JB5drMnsEc4D9eh6tnDMKSSDbzTvGrdaPQRnVFTcW8pI2ZQCWOKF+ZA70OB+qdcLeG

返回数据包特征

Set-Cookie: rememberMe=deleteMe;



六、shiro550和721的区别

1、Shiro550的默认密钥构造恶意的序列化对象，进行编码来伪造用户的Cookie，服务端反序列化时触发漏洞，执行命令。
2、Shiro721的AES加密的key基本上猜不到，由系统随机生成，需要登陆后rememberMe去爆破正确的key，也就是利用有效的rememberMe Cookie值来实现反序列化漏洞的攻击，难度相对较高。

3、一些版本问题

- Shiro1.2.4之前登录时默认是先验证“rememberMe”的值，而不是先进行身份认证，这也是Shiro550漏洞能够利用的原因之一，可以利用伪造rememberMe来绕过身份验证，从而实现未授权访问。
- Shiro1.2.4之后的登录是先进行身份验证，而不是先验证“rememberMe” 所以用户需要知道受害者已经通过登录的验证，并且Shiro框架已经为受害者创建了一个有效的会话，以便攻击者可以利用该会话ID进行身份伪造并绕过Shiro框架的权限控制机制。
- Shiro框架的登录流程也是可以自定义的。