一、介绍

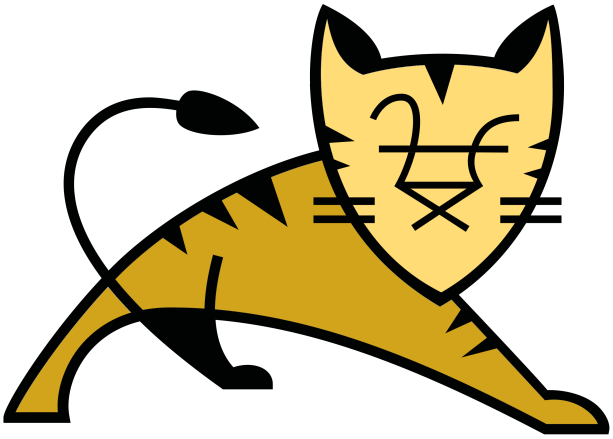
1、简介

```
    Tomcat是Apache基金会下一款轻量级的应用服务器，可实现对JSP的支持
```

2、指纹

```
    端口：8080
```

```
    icon_hash="-297069493"
```



Apache Tomcat

二、漏洞

弱口令

```
# CVE编号
    无
```

```
# 影响版本
    所有
```

```
# 利用方法
    tomcat/tomcat
```

```
# 参考链接
    https://vulhub.org/#/environments/tomcat/tomcat8/
```

反序列化 2020.05.21

```
# CVE编号
    CVE-2020-9484
```

```
# 影响版本
    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
```

```
# 利用方法
    1、使用ysoserial工具包生成反弹shell的payload到/tmp/test.session
    2、curl 请求目标地址，Cookie设置为 Cookie: JSESSIONID=../../../../../tmp/test
```

```
# 参考链接
    https://cloud.tencent.com/developer/article/1665295
```

文件包含 2020.02.25

```
# CVE编号
    CVE-2020-1938
```

```
# 影响版本
    Apache Tomcat 9.x < 9.0.31
    Apache Tomcat 8.x < 8.5.51
    Apache Tomcat 7.x < 7.0.100
    Apache Tomcat 6.x
```

```
# 利用方法
    https://github.com/YDHCUI/CNVD-2020-10487-Tomcat-Ajp-lfi/
    使用py脚本读取WEB-INF/web.xml文件
```

```
# 参考链接
    https://cloud.tencent.com/developer/article/1590029
```

```
# 漏洞分析（问题）
    1、AJP是什么？
        - 定向包协议
        - Tomcat通过8080和8009两个端口与客户端进行通信，两个端口对应两个协议，8080HTTP，
8009AJP协议，AJP协议使用二进制方式传输文本，降低HTTP请求的处理成本
    2、版本特征？
        - Tomcat配置文件conf/server.xml中，定义了8080端口负责处理HTTP协议通信内容，8009
端口负责处理AJP协议通信内容
        - AjpProcessor类用于接收AJP协议通信内容，PrepareRequest用于处理AjpProcessor内
容，将其作为Request的Attribute字段
        - Attribute中有三个属性，include.request_uri，include.path_info，
include.servlet_path，通过向8009端口传入数据即可控制这三个属性
        - 最后根据程序中自身的内容处理方法配合以上三个属性，即可达成文件读取和文件包含目的
```

命令执行 2019.04.16

```
# CVE编号
    CVE-2019-0232
```

```
# 影响版本
    Apache Tomcat 9.0.0.M1 - 9.0.17
    Apache Tomcat 8.5.0 - 8.5.39
    Apache Tomcat 7.0..0 - 7.0.93
```

```
# 利用方法
    1、访问 http://ip:8080/cgi-bin/hello.bat?c:/windows/system32/ipconfig
    2、成功执行ipconfig命令
```

```
# 参考链接
    https://cloud.tencent.com/developer/article/1512468
```

文件上传 2017.09.20

```
# CVE编号
    CVE-2017-12615
```

```
# 影响版本
    Apache Tomcat 7.0.0 - 7.0.79
```

```
# 利用方法
    1、访问tomcat主页，使用PUT方法上传shell.jsp/
    2、访问shell.jsp连接木马
```

```
# 参考链接
    https://github.com/vulhub/vulhub/blob/master/tomcat/CVE-2017-12615/README.zh-cn.md
```

```
# 漏洞原理
    Tomcat的conf/web.xml中，readOnly方法默认为true，表示只读，进制PUT和Delete方法，如果开发人员设置为false，及代表开启PUT和Delete方法，PUT为将请求的实体存储在指定的URIz、Delete为删除，通过PUT发方法即可上传任意木马文件，造成命令执行
```