登录

CVE-2017-12615漏洞复现

Coisini / 2019-07-11 09:13:00 / 浏览数 6367 安全技术 漏洞分析 顶(0) 踩(0)

---

1. 引言

这是一个关于任意文件上传的漏洞，在tomcat中启用put方法会导致任意文件可以上传，从而导致服务器权限被获取。

2. 漏洞介绍

2017年9月19日，Apache Tomcat官方确认并修复了两个高危漏洞，其中就有远程代码执行漏洞(CVE-2017-12615)。当存在漏洞的Tomcat 运行在 Windows 主机上，且启用了HTTP PUT请求方法（例如，将 readonly 初始化参数由默认值设置为 false），攻击者将有可能可通过精心构造的攻击请求数据包向服务器上传包含任意代码的 JSP 的webshell文件，JSP文件中的恶意代码将能被服务器执行，导致服务器上的数据泄露或获取服务器权限。

漏洞危害：泄露用户代码数据，或用户服务器被攻击者控制。
影响范围：Apache Tomcat 7.0.0 – 7.0.79

3. 环境搭建

测试环境：windows 10
Tomcat 7.0.56（Tomcat服务器是一个免费的开放源代码的Web应用服务器。）
Jdk 1.8.0
使用工具：Firefox ，burpsuit v1.7.36，
Firefox: 自由及开放源代码网页浏览器。
Burp Suite: 用于攻击web应用程序的集成平台，包含了许多工具。Burp Suite为这些工具设计了许多接口，
以加快攻击应用程序的过程。所有工具都共享一个请求，并能处理对应的HTTP消息、持久性、认证、代理、日志、警报。在一个工具处理HTTP请求和响应时，它可以选择

4. 漏洞代码分析

通过阅读conf/web.xml文件，可以发现：默认 readonly为true，禁止HTTP进行PUT和DELTE类型请求:

```
<!--    readonly      Is this context "read only", so HTTP          -->
<!--                  commands like PUT and DELETE are              -->
<!--                  rejected?  [true]                             -->
```

先知社区

当web.xml中readonly设置为false时可以通过PUT/DELETE进行文件操控，漏洞就会触发。
这个CVE漏洞涉及到 DefaultServlet，DefaultServlet作用是处理静态文件，同时DefaultServlet可以处理PUT或DELETE请求，默认配置如图2：

```
<!-- The mapping for the default servlet -->
<servlet-mapping>
    <servlet-name>default</servlet-name>
    <url-pattern>/</url-pattern>
</servlet-mapping>

<!-- The mappings for the JSP servlet -->
<servlet-mapping>
    <servlet-name>jsp</servlet-name>
    <url-pattern>*.jsp</url-pattern>
    <url-pattern>*.jspx</url-pattern>
</servlet-mapping>

<!-- The mapping for the SSI servlet -->
```

可以看出即使设置readonly为false,默认tomcat也不允许PUT上传jsp和jspx文件，因为后端都用org.apache.catalina.servlets.JspServlet来处理jsp或是jspx后缀的请求，而PUT类型的操作, 所以可知PUT以及DELTE等HTTP操作由DefautServelt实现。因此，就算我们构造请求直接上传JSP webshell显然是不会成功的。该漏洞实际上是利用了windows下文件名解析的漏洞来触发的。根本是通过构造特殊后缀名，绕过Tomcat检测，让Tomcat用DefaultServlet的webshell文件。

具体来说，主要有三种方法：

shell.jsp%20
shell.jsp::$DATA
shell.jsp/

本次测试，使用第一种方法，可成功实现上传，并取得WebShell。

5. 漏洞复现

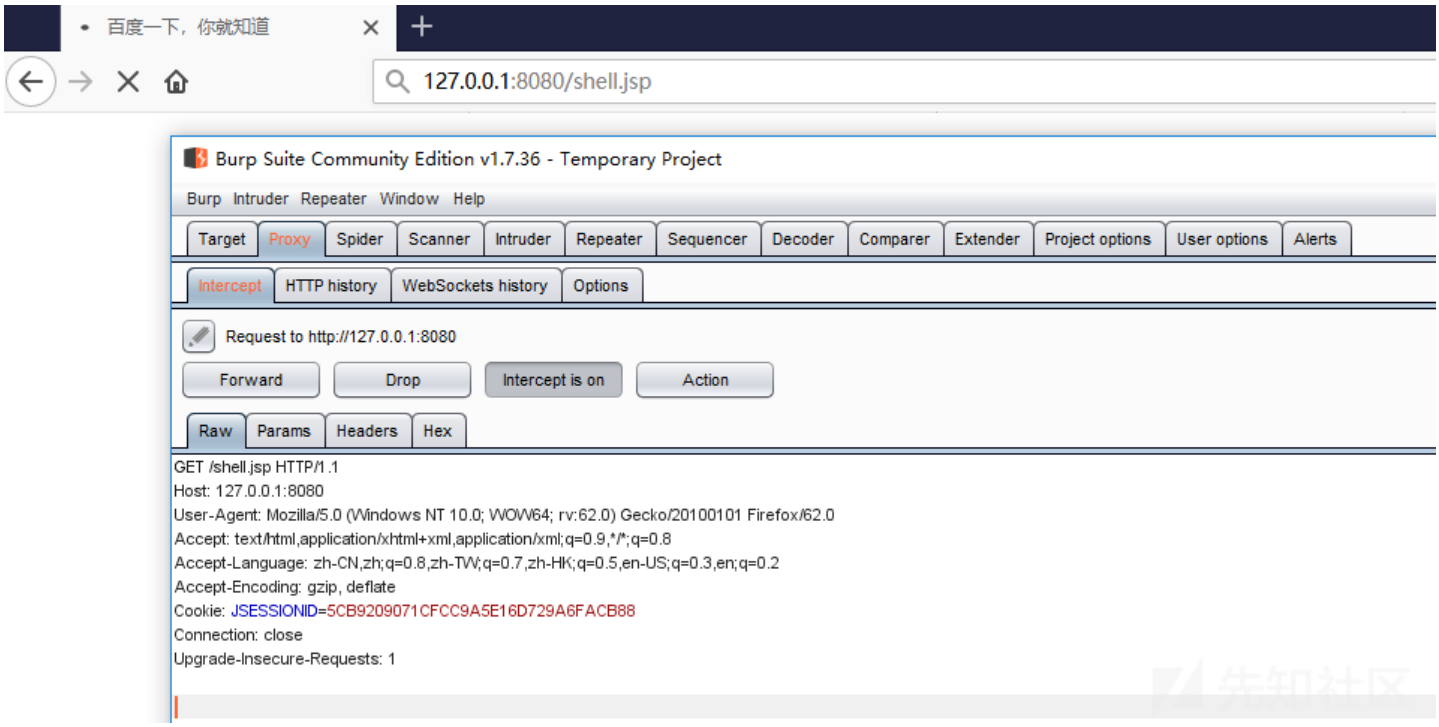
根据描述，在Windows服务器下，将readonly参数设置为false时，即可通过PUT方式创建一个JSP文件，并可以执行任意代码。

Tomcat7中readonly默认值为true，手动将其改为false，在conf/web.xml中手动添加红色方框类内容，如图3：

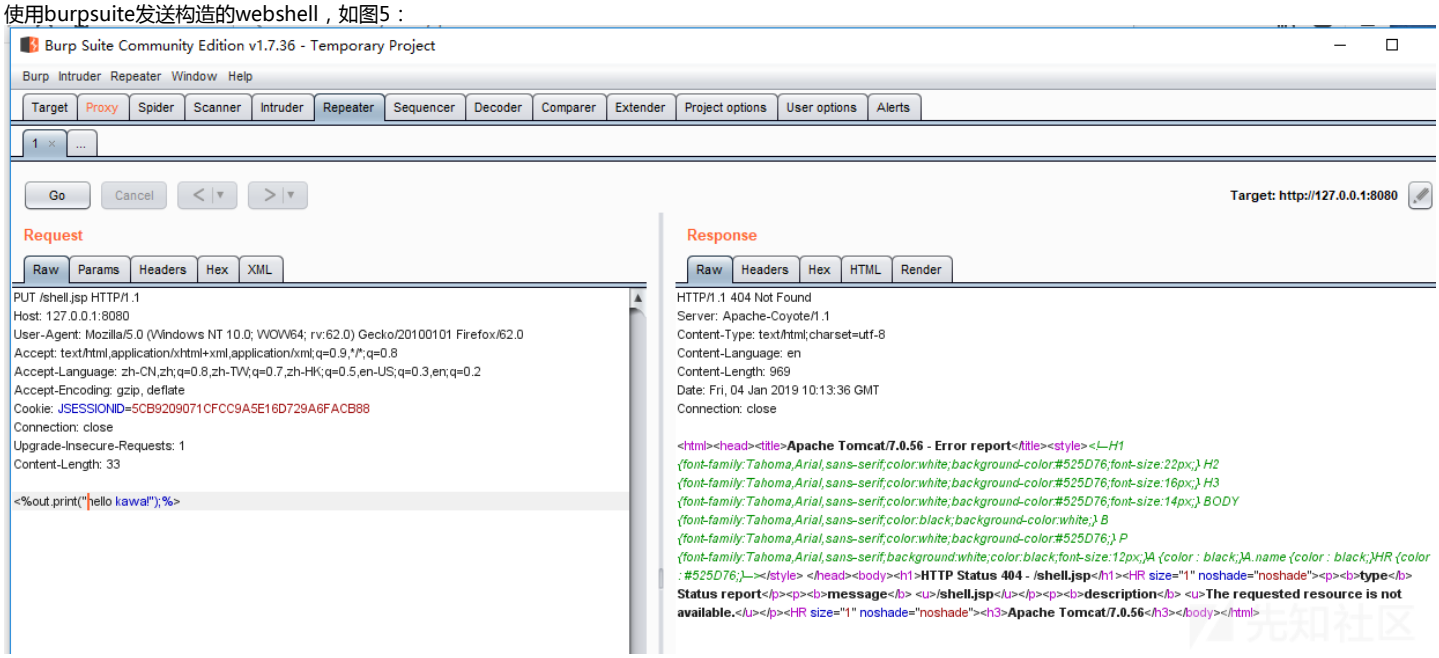
```
<servlet>
    <servlet-name>default</servlet-name>
    <servlet-class>org.apache.catalina.servlets.DefaultServlet</servlet-class>
    <init-param>
        <param-name>debug</param-name>
        <param-value>0</param-value>
    </init-param>
    <init-param>
        <param-name>listings</param-name>
        <param-value>false</param-value>
    </init-param>
    <init-param>
        <param-name>readonly</param-name>
        <param-value>false</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
</servlet>
```

设置完成后，启动Tomcat，利用PUT请求创建文件。

构造webshell请求，使用burpsuite抓包，如图4：

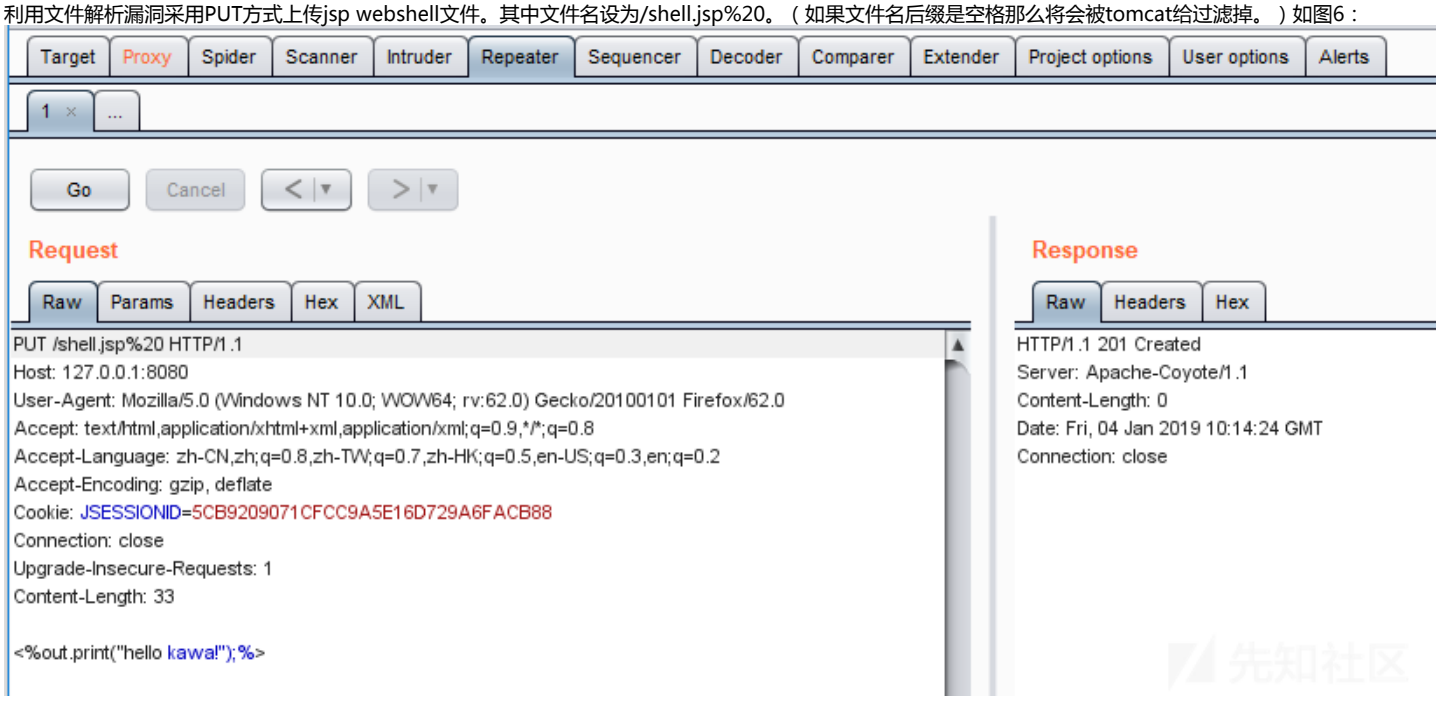


使用burpsuite发送构造的webshell，如图5：

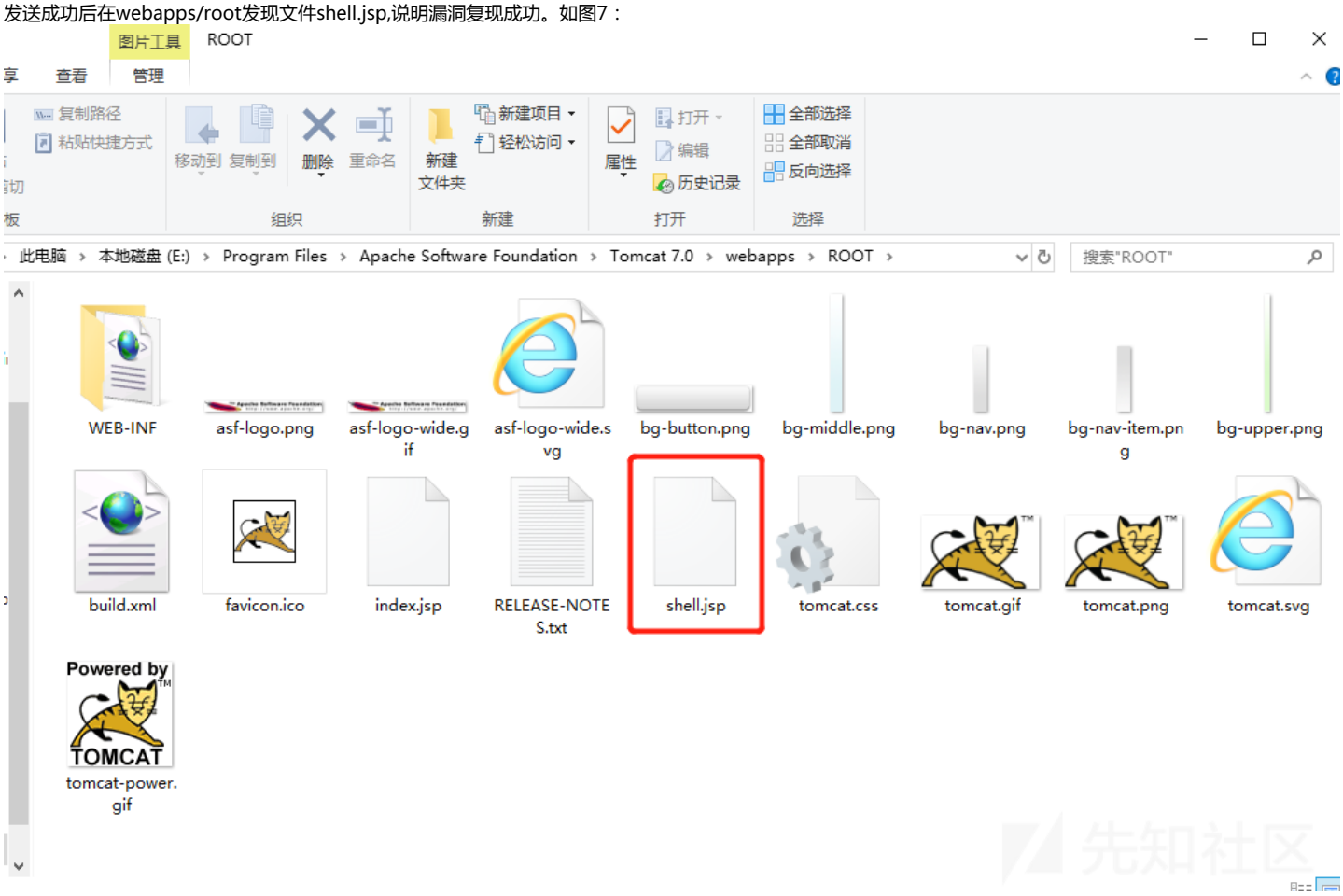


提示404，请求测试结果表明了猜测结论是正确的。JspServlet负责处理所有JSP和JPSX类型的动态请求，不能够处理PUT方法类型的请求。

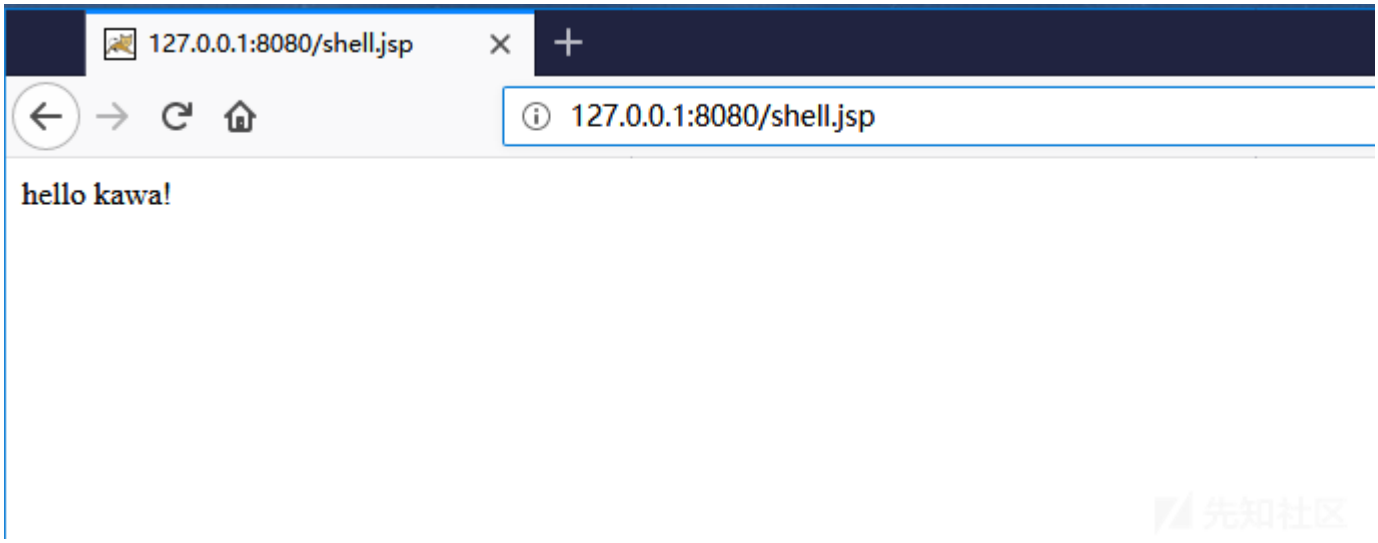
利用文件解析漏洞采用PUT方式上传jsp webshell文件。其中文件名设为/shell.jsp%20。（如果文件名后缀是空格那么将会被tomcat给过滤掉。）如图6：



发送成功后在webapps/root发现文件shell.jsp,说明漏洞复现成功。如图7：



访问发现可以正常输出，如图8：



6. 修复方案

1、配置readonly值为True或注释参数，禁止使用PUT方法并重启tomcat。
注意：如果禁用PUT方法，对于依赖PUT方法的应用，可能导致业务失效。

2、根据官方补丁升级最新版本。

点击收藏 | 0 关注 | 1
上一篇：某info <= 6.2.0前台任... 下一篇：以太坊随机数安全全面分析（一）
1. 0 条回复
   - 动动手指，沙发就是你的了！

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板