Java Web 漏洞生態食物鏈

r4bb1t_ / 2016-12-08 05:34:55 / 浏览数 4102 安全技术 漏洞分析 顶(0) 踩(0)

本文作者Orange，原文地址 <http://blog.orange.tw/2016/12/java-web.html>

> 本来这篇文章叫做HITCON CTF 2016初赛出题小记的，可是摆着摆着就两个月过去惹～
转来写写跟Java有关的东西XD

关于序:

今年五六月的时候，看到某个曾经很多人用但快停止维护的Java Web Framework弱点的修补方式感觉还有戏所以开始追一下原始码挖0-Day，顺便整理一下Java Web相关弱点一 觉得有趣。

通常自己在外面演讲时对于Web Security的分类中大致可分为三个世界:

- File-Based 的世界，一个档案对应一个入口点如经典的ASP, PHP, ASPX 等
- Route-Based 的世界，一个路径对应一组函数(功能)如经典的Rails, NodeJS, Django 等
- Java 的世界，Java 的世界极其复杂自成一格独立讨论

当然三种分类并不是独立开来，如常见PHP MVC 用Rewrite 将File-Based 伪装成Route-Based 的还是有可能有File Based 的特性，Java 世界中还是可能出现File-Based / Route-Based 常见问题等等等

Java 可以独树一格自成一圈自然有它的道理在! Java Web 相关攻击手法一直以来都是扑朔迷离，生态复杂出过的弱点又多(想想Struts2 :P)，偶尔爆出一个很严重的漏洞才又开始有人关注，也很难有一个脉络所以一直以来都没有看到有个很好的整理!
(当然国外还是有一些在这个领域耕耘已久的研究人员如@pwntester @meder @ gebl @frohoff @空虚浪子心等等等)

这也促使我在这下半年开始来好好研读一下关于Java Web 的生态以及弱点漏洞的整理。

从2013 年开始接触CTF 到现在，在Web 分类中所看到关于Java 的题目真的少之又少，最常见的通常是PHP、其次Python、偶尔来点Ruby / JavaScript / Go 等等...

虽然PHP最大宗很合理(现实网站常见、新手容易入门、语法特性容易写出问题)，不过身为稳定度高，在大型网站、银行、政府网站常见的Java几乎没有就很不合理了。

最后是，PHP的梗几乎都被玩完了，除了老梗外几乎都是基于PHP核心问题的特性，该是换点新梗的时候!

关于 Java:

Java Web的生态各种复杂，从最底层的JVM到Web Container到上层的Web Framework，在Java的世界中就像是一种原料什么都可以靠它堆塑起来，而不像以往的PHP只须要顾好应用层就好!再加上在Java生态中很喜欢引用来、引用去，串来串去的Admin Console这个应用来当例子的话则是:

JBoss Admin Console使用Seam Framework这个框架所开发的网页应用，而Seam Framework使用了JSF(Java Server Faces)的架构，Java Server Faces为一个Java EE的标准，基于这个标准上的实作较知名的共有两套：

- Apache 所实作的MyFaces
- Oracle 所实作的Mojarra

而为了让JSF 实作更为好用又在其上引申出了一些方便JSF 使用的Framework 如Richfaces
而Seam 则是使用基于Richfaces 上的实作，所以整个生态链为
`JVM -> JBoss -> Mojarra -> Richfaces -> Seam Framework -> JBoss Admin Console`

整个生态链串来串去只要其中一个出现漏洞则最上层的应用皆会有问题

如:
CVE-2010-4476，JVM 浮点数解析DoS 漏洞，只要以上处里的过程中出现类似 Double.parseDouble("2.2250738585072012e-308") 的状况就可以达成DoS 效果
Web Container /Application Server 弱点，这个不细数，JBoss, Tomcat, GlassFish, Weblogic 或是Websphere 都出过很多洞XD
JSF 实作漏洞，可以看下面举的例子
Framework 本身漏洞，如 CVE-2010-1871 - Seam EL注入漏洞导致远端代码执行
网页应用开发者本身自己写出来的洞

举个例子:

在研究Java网页应用弱点的时候发现到了CVE-2010-1871这个好玩的东西
漏洞详情可参阅《JBoss Seam Framework remote code execution》

虽然Seam Framework 已经是过时的产品了不过身为弱点研究人员感兴趣的是漏洞的成因跟如何修补，在研究漏洞如何修补的同时顺便对Seam Framework 的原始码审查一下，发现在Seam Framework 2 的这个分支中最新的版本为2.3.1.Final，在前文有提到，Seam 所使用的是基于Mojarra 实作的Richfaces，这代表只要Mojarra 或是Richfaces 有弱点，则可影响到所有使用Seam2 所撰写的应用程式。

再仔细观察一下Seam 2 所使用到的函示库
lib/richfaces-core-impl.jar - 根据MANIFEST 内容显示版本为Richfaces 4.3.3.Final
lib/jsf-impl.jar - 根据MANIFEST 内容显示版本为Mojarra 2.1.7

最新版本的Seam 2使用的居然不是最新版本的第三方函示库!透过寻找上面两个函示库出现过的漏洞会发现在2013年出现的CVE-2013-3827
详情可参考《Two Path Traversal Defects in Oracle's JSF2 Implementation》

所以基于CVE-2013-3827上，将原本的PoC修改一下就可以完美的重现在最新版本的Seam Framework 2上面。

上面这个例子只能做到读取敏感档案，接下来再一个远端代码执行的例子:
同样也是使用到旧版本Richfaces的CVE-2013-2165，开发者根本不会知道多了一个/a4j/的url-pattern，漏洞会从url中直接将参数代入readObject执行，刚好可以搭上最近
Deserialization风潮达成远端代码执行!

关于这样应用建立在前面的基石，当基石出问题时整个倒下来的情形在Java世界中并不少见，最近很夯的Java Deserialization相关问题也有类似这样的生态链，还不觉得Java的世界很有趣吗? XD

好惹，差不多降!
这只是个随笔就先富坚，有空的话再多写一点好惹

小结:

由于生态完整并要注重系统稳定，大部分使用Java当成网站开发的都是些大型网站或注重稳定性的网站如金融业或是政府机构，并且为求稳定通常不会频繁的更新系统，所以

如在研究时顺便观察到Apple的网站有使用Mojarra的特征，2013年的漏洞丢下去居然还可以用Java Web Framework相关的漏洞真的满边缘人的XDD
`https://???.apple.com/???/javax.faces.resource.../WEB-INF/web.xml.jsf`

出来混迟早要还的，不然技术债会越累积越多，接下来有空的话应该会尝试检视Spring Framework Source Code，毕竟也是现在的主流!
至于S什么什么T什么什么2的就不用多讲了XD

题外话1:

题外话是，在Review 完Seam Framework 后回报了几个漏洞给security@jboss.org ，不过回覆则说因为Seam 只能在JBoss EAP 7 下使用，而JBoss EAP 也即将在2016/11 月停止维护，所以除了重要或是严重风险的漏洞外皆不修复。

所以你知道的，现在用Seam写的网页都可以
嗯，真棒XD

题外话2:

Apple敏感档案存取的漏洞是新训进去前发现的，出来要回报时就发现不能用不知道是修掉还怎样QQ
本来想直接公开不过还是在发布文章前最后一秒把关键字码掉惹

上一篇：APP漏洞扫描器之未使用地址空间随机化 下一篇：初探CSPBypass一些细节总结

1. 1 条回复

gr3g_ 2016-12-12 04:00:27

这个文章很棒，梳理的很清楚，java的框架依赖复杂的让人蛋疼。

0 回复Ta

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