安全小课堂五十六期【二次注入漏洞解析】

京东安全应急响应中心 2017-05-05

二次注入漏洞是一种在Web应用程序中广泛存在的安全漏洞形式。相对于一次注入漏洞而言，二次注入漏洞更难以被发现，但是它却具有与一次注入攻击漏洞相同的攻击威力。JSRC **安全小课堂第五十五期和五十六期**，我们来聊一聊二次注入漏洞原理解析，两期小课堂邀请到了**王柯**、**Chu**、**重道远zxx**师傅进行分享。感谢JSRC白帽子们的精彩讨论。

简要描述二次注入漏洞?

京安小妹

简单的说，二次注入是指已存储（数据库、文件）的用户输入被读取后再次进入到 SQL 查询语句中导致的注入。

可能每一次注入都不构成漏洞，但是如果一起用就可能造成注入。

讲师：王柯、Chu

二次注入漏洞存在的原因?

京安小妹

个人觉得说到底还是信任的问题
开发者可能不信任直接来自用户的数据，对其进行转义后存储。但对于已存储的数据却过于信任，数据未经过滤、转义被取出后放入了 SQL 语句中，自然就导致了注入。
也有一些是因为开发者对语言中的函数不理解所导致的，比如经典的 PHP is_numeric。
如果是用户直接输入的内容，那么大家都有个安全意识，认为可信度低，于是做了转义处理,但是对于从数据库取出来的数据，安全意识就没这么高了。其实也有可能是用户间接输入的。

讲师：王柯、Chu

二次注入漏洞有何危害?

京安小妹

二次注入的危害就是sql注入的危害。但是由于这类漏洞隐蔽性较好，不易被发现，所以更危险。不过还好，它的存在的数量肯定远小于直接性的sql注入。因为这个攻击位于db上，db上能做的就那么多么。常规注入导致的危害他也不例外。

讲师：王柯、Chu

二次注入漏洞与普通注入漏洞的区别（特点）？

京安小妹

二次注入是sql注入的一种，但是比普通sql注入利用更加困难，利用门槛更高。
一个是输入数据流向。普通注入数据直接进入到 SQL 查询中，而二次注入则是输入数据经处理后存储，取出后，再次进入到 SQL 查询。
渗透过程中，流程越是复杂，遇到的不确定因素就越多，所以成功率也就月底,例如，如果第二次注入必须由管理员在后台来被动触发，这个条件就很苛刻了~

讲师：王柯、Chu

白帽子提问:二次注入漏洞都是手工发现的的吗 有没有相关的检测软件或者小工具推荐呢?

白盒的话，开源的工具没有发现特别好的。因为想要挖掘二次注入首先要对应用有一个完整的理解，对不同功能间的关系有一定的理解。
二次注入虽然是sql注入的一种，但是他的特性里面又带一点逻辑的意思。所以自动化的扫描是很难发现的，基本都是靠人工。

讲师：王柯、Chu

挖掘二次注入漏洞时应该注意什么?

京安小妹

注意点就是细心和耐心，在回溯数据输入时不应该忽略来自数据库、文件等的输入，而应该回溯这些输入的初始来源，进一步判断是否可控。还要就是对一些函数的理解、关注，比如之前说的is_numeric，观察到相应函数后应该去判断下是否存在勿用的情况。
对应用的理解+细心+耐心；敏感函数是否存在勿用。

讲师：王柯、Chu

白帽子提问:二次注入，不通过代码审计黑盒可以测出吗?

输入点你打入payload，能否完整遍历接口，找到输出的点，这个是黑盒的问题。
很难通过普通的黑盒扫描找出来。不过，如果是报错注入的只能能爆出sql语句的话，能用http的响应里面找到一些蛛丝马迹。
扫描器扫存储式xss的时候，就是像初老师说的这些，通过遍历每个接口提交不同标识的payload，然后用爬虫寻找输出。

讲师：王柯、Chu

白帽子提问:二次注入好像php上碰的多，Java代码出现的多吗？？

java只要是拼接执行 一样会有的。
java很少碰到拼接的，大部分都是预编译了，除了开发人员自己的问题外,特别是跨语言的应用，比如后台java ，前台php，很容易导致问题。

讲师：王柯、Chu

白帽子提问:有没有挖掘二次注入漏洞的小技巧呢?

一般挖掘注入我习惯在查询函数上打点，断点或者log都可以。然后在回溯代码的过程中观察log，很方便。
或者就是硬着头皮审计喽。

讲师：王柯、Chu

企业应该如何防御二次注入漏洞?

京安小妹

解决SQL注入最推荐的方法还是预处理+数据绑定。
另一个防御的点就是 对输入一视同仁，无论输入来自用户还是存储，在进入到 SQL 查询前都对其进行过滤、转义。
对于二次注入这种小众的漏洞，一般没有专门针对的方案，只能从流程上进行优化，例如做代码审查的时候禁止开发用拼接的方式执行sql。很多开发他不懂安全也不懂攻防，到写出来的代码合规合流程，就是没漏洞。回答完毕。

讲师：王柯、Chu

请两位师傅谈一谈，二次注入漏洞检测的一般流程是怎么样的?

京安小妹

这个百盒比黑盒要更match容易些。白盒的话最好倒着看，先看有没有拼接，输入点是不是入库了。
白盒的话，和其他漏洞流程一样，只是要多关注一下，写入到数据库、文件、缓存中的内容，这点困难会比较麻烦match，倒着跟会比较好点。在有源码的情况下，其实之前看过有的文章说使用动态静态结合，先分析哪些数据库的列是可控的，可以减少一些影响因素。

纯黑盒的话，主要还是输入一点标记内容，然后靠看和分析。

讲师：王柯、重道远zxx

那二次注入漏洞挖掘的难点在哪里呢?

京安小妹

黑盒的时候比较麻烦，不像存储的XSS，还可以有个盲打。在没有报错的情况下，发现了可能二次注入的的地方，测试一个payload，也要花很多心思。

白盒的时候也比较麻烦，因为输入和输出可能是异步的，比较难去关联数据的输入和输出，所以不好检测。它不像普通的注入，请求和响应是立即匹配的有可能输入以后，两年后才触发注入。

讲师：王柯、重道远zxx

白帽子:我还真遇到过，一年多后才触发的存储xss。

关于二次注入漏洞是否有自动化检测工具?

京安小妹

我没见过有专门检测二次注入的，不过如果有兴趣我们可以写一个。^_^

最近没有关注，RIPS的作者发了一篇文章是讲检测二次注入漏洞的，发表时间是2014年。不知道RIPS的下一代版本是不是已经支持了这个特性。《Static Detection of Second-Order Vulnerabilities in Web Applications》，有兴趣的可以看一下。
我们skywolf也是可以检测二次注入的，但目前误报较高，还需要优化。

讲师：王柯、重道远zxx

一般什么样的功能容易出现二次注入漏洞?

京安小妹

遇到一个印象比较深的就是打日志的时候。存日志时，读取了一些数据库里的信息，比如用户名等，然后又存储了一次。
还有跨程序的数据传递，程序A处理完后存到数据库，程序B去读取，没有进行过滤。
比如说Chu提到的例子，前台是Php，后端程序是Java，可能又不同的开发来实现，信息不对称，后端认为数据是无害的，没有处理。

讲师：王柯、重道远zxx

二次注入漏洞是否可能存在像普通注入一样修复后被绕过的情况？如何防止这种情况的发生？

京安小妹

当然可能，二次注入的修复可以把它当做普通注入一样。如果也来个关键字过滤什么的，肯定还是能被绕过，只是测试门槛高了。
抛开二次注入的“第一次”，把从数据库输出的内容当GPC传递过来一样来处理，后面完全可以跟sql注入修复一样，普通注入怎么修，二次注入就怎么修。

对，例如它对到双引号转义了，但是拼接到limit或者order by 或者其它位置后面，一样会被注入的。

讲师：王柯、重道远zxx

本期JSRC 安全小课堂到此结束。更多内容请期待下期安全小课堂如果还有你希望出现在安全小课堂内容暂时未出现，也欢迎留言告诉我们。

安全小课堂的往期内容开通了自助查询，回复“安全小课堂”或者点击阅读原文进行查看。

简历请发送：cv-security@jd.com

微信公众号：jsrc_team

新浪官方微博：京东安全应急响应中心