安全小课堂第九十五期【web漏洞之XSS漏洞挖掘】

京东安全应急响应中心 5月15日

XSS是一种经常出现在web应用中的计算机安全漏洞，它允许恶意web用户将代码植入到提供给其它用户使用的页面中。比如这些代码包括HTML代码和客户端脚本。
JSRC **安全小课堂第九十五期**，邀请到**gainover**作为讲师就**web漏洞之XSS漏洞挖掘**为大家进行分享。感谢白帽子盆友的精彩提问与互动~

常见的XSS我们都知道分类为反射型，存储型，Dom型，之前还听说过一个UXSS，能给大家介绍介绍吗？要怎么去发现这类XSS

京安小妹

gainover：
其实这个uxss虽然名字里有xss，但是和我们平时挖的xss是两回事，uxss属于浏览器的漏洞，xss则是web应用的漏洞。对uxss，虽然有找过一些，基本也是国产手机浏览器和国产浏览器的，但是对于ie，chrome这些的uxss，自认不会。
我大概归为四个小点：一是从代码层面白盒挖掘，需要对开源的浏览器代码非常熟悉才行，https://www.seebug.org/search/?keywords=uxss&category=&level=all，比如seebug这个list里的；第二，黑盒的，比如ie的uxss，没有源代码的情况，https://blog.innerht.ml/ie-uxss/，比如这个是近年比较经典，也实用的一个；印象中还有另外一个与 activex htmlfile 有关的，这种漏洞一般是有历史漏洞参考，就是在这块曾经出现过问题，然后发现者功底深厚，积累了许多小的trick，形成完整的链条；还有就是浏览器内置的功能页，浏览器插件都是会导致uxss的，以前测试国内浏览器有这种情况。

讲师

测试XSS最常碰到的是防火墙，有什么绕过方式？

京安小妹

gainover:

这个问题其实有点泛了，实际挖掘过程中的话，还是得结合输出点和waf本身的检测规则具体而论。本身后端waf是个黑盒，如果确定有waf，基本就开始了一个黑盒摸索的过程，我们需要的是摸清楚它的规则，比如是过滤了关键符号，比如<>，还是过滤了特定标签或是特定属性，是多组正则规则，还是仅仅是判断关键字包含，接着就是靠经验或者fuzz的手段来尝试绕过规则。

讲师

我们知道大部分人对XSS的印象是弹框和盗取cookie，能讲讲XSS更多的攻击场景吗？比如在内网渗透中的作用？

京安小妹

gainover:

按正常推理的话，内网中的作用应该是内网弹窗吧哈哈哈，开个玩笑，其实xss能做什么，取决于你拿xss下一步做什么，xss只是给你了执行代码的能力，实际利用的话，比如ctf里有过的xss打redis，内网的信息收集，比如端口开放情况，应用开放情况，再比如借由浏览器做跳板，结合浏览器漏洞，远控员工机器。还有些简单的场景，xss+phishing钓密码啥的。

我们都知道XSS盲打，甚至说类似beef这种XSS攻击框架，能讲讲其原理吗？

京安小妹

gainover：

拆开讲，盲打就是我们不知道到底打不打的到，不知道目标代码上下文到底是啥，所以通过考虑各种可能性，构造出一个大多数时候都可用的xss代码来进行测试。没记错的话，好像是从乌云上火起来的，当时是个雪球网还有百度一个站的案例，随之而来的还有xss platform，beef作为国外知名的xss利用框架，其实我只看过别人写的教程，但是，实时交互式的xss平台还是非常实用的，既然不愿意用beef，所以我们自己团队是有开发类似功能的平台。从原理上，我觉得有几个关键点，一是实时，对于现代浏览器，基于websocket来实现，对于老一些的浏览器，有flash插件的，可以基于flash的socket，再老一些的，可以ajax polling ，甚至iframe polling；二是 各类利用代码的模块化问题，这包括比如模块代码能否重用，模块的参数等等；三是beef里会有与metasploit结合，发挥浏览器漏洞的威力，不论是用beef，还是自己开发的，对于使用者来说，我觉得更多还是熟悉什么攻击场景能用到xss去解决什么问题吧。

讲师

针对现在以各种如react之类框架开发的Web应用，挖掘XSS有没有什么不同，有没有什么技巧?

京安小妹

gainover

现在框架很多，封装的越来越厉害了，比如angular，react啥的，我前些天拿到一份其他前端写的东西，我都不知道从哪下手去启动它，所以其实要想从高度封装的代码里去找问题。要么是深入使用它，熟悉它，要么就是实际找漏洞时候，刻意避开它。比如前些天，在tsrc群里看到有人说某某应用是react写的，后来我也去看了下，说实话，我都没注意哪里有react的东西，避开了它一样也会发现还是有漏洞的。当然有时候避不开，那我们还是得如前所说，去熟悉它，然后了解它出现过的安全问题。当然，这些个框架我自己没用过，所以也仅仅限于从公开渠道学习别人的好文，比如angular的这篇https://portswigger.net/blog/xss-without-html-client-side-template-injection-with-angularjs再比如这篇讲script gadget的https://www.owasp.org/images/3/32/OWASP_BeNeLux-Day_2017_Bypassing_XSS_mitigations_via_script_gadgets_Sebastian_Lekies.pdf都很nice。还有一些公开漏洞的学习资源，比如hackerone，我印象中看到的第一个react的xss的漏洞就是这个上面看的，时间久了找不到链接了，这块我实在没什么仔细研究过，就这么过了，多包涵。

讲师

除了传统的关键词检测之类的，对于XSS的防御有没有一些新思路？

京安小妹

gainover:
防御这块，传统的关键词检测一般是由waf或全局过滤来做的，加上浏览器内置的filter越来越强大，大多数情况下其实还是够用了。要说增强的话，一个是waf上的改进，比如长亭他们的 xsschop； 二是加上一个前端的检测机制，用js检测dom xss，如果有挖腾讯的xss的应该会熟悉这个；第三点就是csp 以及它的report机制也很实用。对于第二点，前端js代码的检测，可以弥补后端无法发现的dom xss，或者是hook xss测试和xss利用时经常使用的函数。当然，我没在甲方呆过，这些搜集的数据能有多少，误报多不多，没实际体会过。

讲师

作为一个公司的技术人员又该如何去进行XSS漏洞的防御？

京安小妹

gainover:
对于简单的输出点，了解什么上下文该过滤什么，并且给出正确的过滤机制,了解什么时候用什么函数更安全，避免使用有潜在安全问题的函数。这个说起来是轻松的，我觉得这种应该有个checklist之类的学习。对于开发来说，实际上是一个认知对抗与提升的过程，相信很多人都有漏洞反复修复反复提交的过程，实际上就是开发在拉近和你的思维差距，最终你们认识一样了，漏洞就修好了。

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，点击菜单栏进入“安全小课堂”即可浏览。

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