XSS漏洞挖掘与安全防护

MIMAZ.ORG

宋健

宋健

目前在读辽宁科技大学

ID:宋宋宋

维护密码站相关项目的正常运作，包括：ManyScan漏洞扫描器，密码站社工数据查询平台…

XSS是一种经常出现在web应用中的安全漏洞，它允许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。为了不和层叠样式表(Cascading Style Sheets, CSS)的缩写混淆，故将跨站脚本攻击(Cross Site Scripting)缩写为XSS。恶意攻击者往Web页面里插入恶意Script代码，当用户浏览该页之时，嵌入其中Web里面的Script代码会被执行，从而达到恶意攻击用户的目的。这种类型的漏洞由于被黑客用来编写危害性更大的网络钓鱼(Phishing)攻击而变得广为人知。对于跨站脚本攻击，黑客界共识是：跨站脚本攻击是新型的“缓冲区溢出攻击“，而JavaScript是新型的“ShellCode”。 在2007年OWASP所统计的所有安全威胁中，跨站脚本攻击占到了22%，高居所有Web威胁之首。

1. UXSS(Universal XSS)：

UXSS有着基本XSS的特点，利用漏洞，执行恶意代码，UXSS是利用浏览器或者浏览器扩展漏洞来制造产生XSS的条件并执行代码。 UXSS不需要一个存在问题的页面来触发攻击，它可以渗透到没有问题的页面，导致安全问题。UXSS攻击不需要页面本身存在漏洞，同时可能访问其他无漏洞页面。

2. 反射型XSS(Non-Persistent XSS)：

客户端或服务端代码开发不严谨等问题导致存在漏洞的目标网站或者应用程序。攻击的先决条件是页面存在漏洞，而影响往往也围绕着漏洞页面本身的用户会话。该型漏洞往往不持久，需要欺骗用户自己去点击链接才能触发XSS代码（服务逻辑不允许这样的页面出现）。

3. 存储型XSS(Persistent XSS)：

这种XSS是最为广泛而且有可能影响到Web服务器自身安全的漏洞，存储型XSS会把用户输入的内容保存在服务器端。具有很强的稳定性。

4. DOM-XSS(DOM Based XSS)：

从效果上来说也是反射型XSS，单独划分出，是因为成因较为特别，通过修改页面的DOM节点形成的XSS，进而单独拿出分类。

5.Self-XSS(Self-XSS)：
自身的XSS攻击，攻击的受害者不小心运行恶意代码在自己的Web浏览器，从而成功攻击到目标。这种类型漏洞单凭自身很鸡肋，一般配合其他漏洞才能发挥其危害，如CSRF+Self-XSS

6.POST-XSS(POST-XSS):
通过POST请求，返回恶意内容导致的XSS问题。

7.MXSS(Mutation-based Cross-Site-Scripting)：
翻译为突变型XSS，也叫MXSS。
攻击较为隐蔽，常规的XSS过滤器很难防止此类攻击。

其他类型XSS暂不垒述。

1.模拟用户操作类：
a.攻击数据，增删改查企业敏感数据。
b.非法支付/转账。
c.发送消息，发送邮件。

2.攻击类：
a.盗窃企业重要的资料。
b.盗取管理员cookie，进入后台脱取敏感数据。
c.挖矿。

3.无聊类：
a.弹窗。
b.蠕虫。
c.自动跳转。

XSS漏洞挖掘与安全防护
公开的安全事件(参阅公开漏洞库和SRC公开的问题点)

XSS相关漏洞：（*云镜像）

2016-05-09	某站存储型xss漏洞可获取超级管理员权限	XSS 跨站脚本攻击
2016-05-05	又一XSS被广泛用于黑产	XSS 跨站脚本攻击
2016-05-03	某地区分站Flash逆向分析存安全隐患可进行XSS利用	XSS 跨站脚本攻击
2016-05-03	某插件漏洞导致域名xss	XSS 跨站脚本攻击
2016-04-28	APP安全之一处无效xss爆破导致www后台被登录(100万设备/150万用户/1000万微信用户/核心功能可修改用户密码)	XSS 跨站脚本攻击
2016-04-26	某系统存储型XSS漏洞(已登录系统)	XSS 跨站脚本攻击
2016-04-23	头条页面存在XSS漏洞	XSS 跨站脚本攻击
2016-04-21	APP安全之一处鸡肋无权限xss可操控27w用户总交易金额22亿(大量持身份证照片)	XSS 跨站脚本攻击
2016-04-14	网某重要后台XSS(已打管理Cookie并登录)	XSS 跨站脚本攻击
2016-04-12	某信息查看插件信息输出未处理可导致XSS（Zone测试为例）	XSS 跨站脚本攻击
2016-04-12	储存型XSS（过滤了尖括号/圆括号/单引号等字符下的利用技巧）	XSS 跨站脚本攻击

首页 前一页 1 2 3 4 5 后一页 尾页

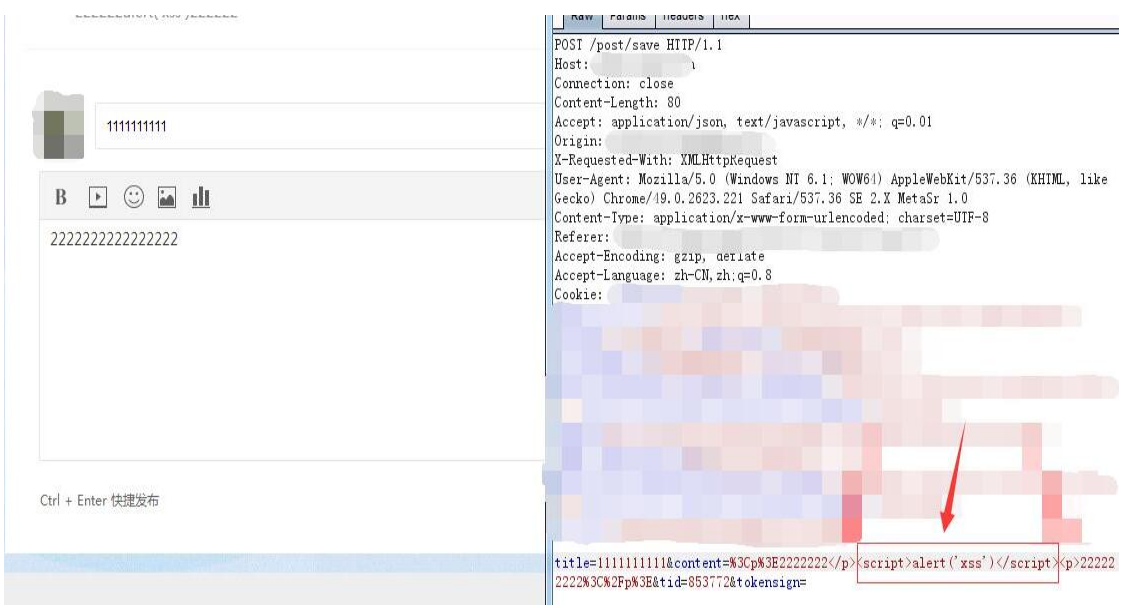
同程安全应急响应中心：https://sec.ly.com/bugs

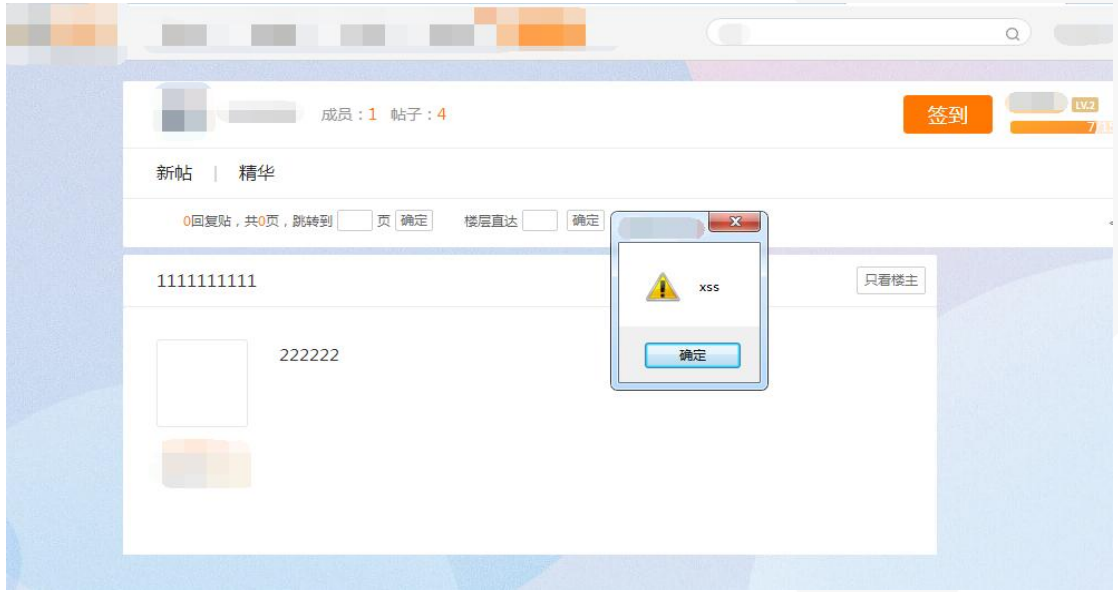
安全实例：

以A论坛xss安全进程为例，讲解XSS测试的一些小技巧：

论坛的评论输出点，经久不息存在问题，最容易测试的位置，也是最不可能存在问题的位置。

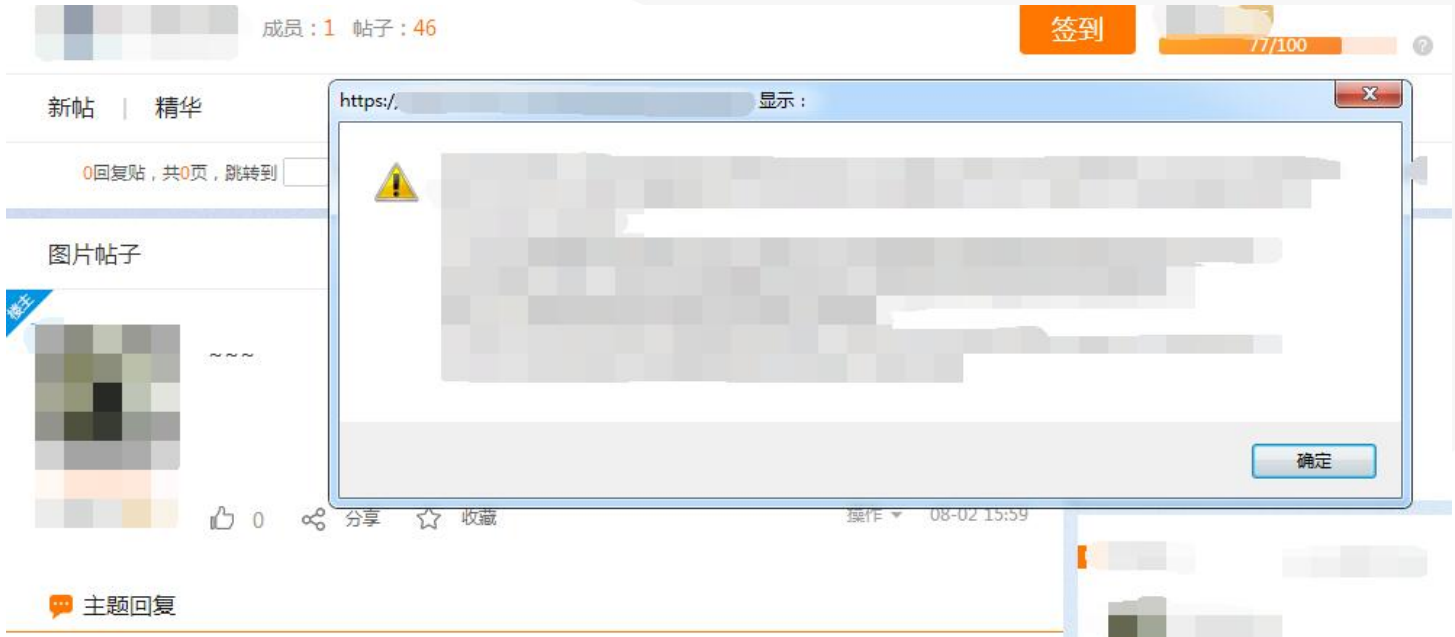
A论坛发帖导致的XSS问题：





安全实例：

修复后，在web端无法进行html代码注入，那么考虑其他输入点，例如APP端进行尝试输入：
A论坛APP端发帖导致的XSS问题：



安全实例：

这次A论坛加强了防御，对APP和web端输入均进行了黑名单过滤，尝试考虑绕过：

A论坛发帖，绕过黑名单规则，导致的XSS问题：

发帖的时候，遇到敏感字符会有[redacted]防火墙拦截

绕过：

POC："><s cript>al ert(1);</s cript>

注意空格！！！

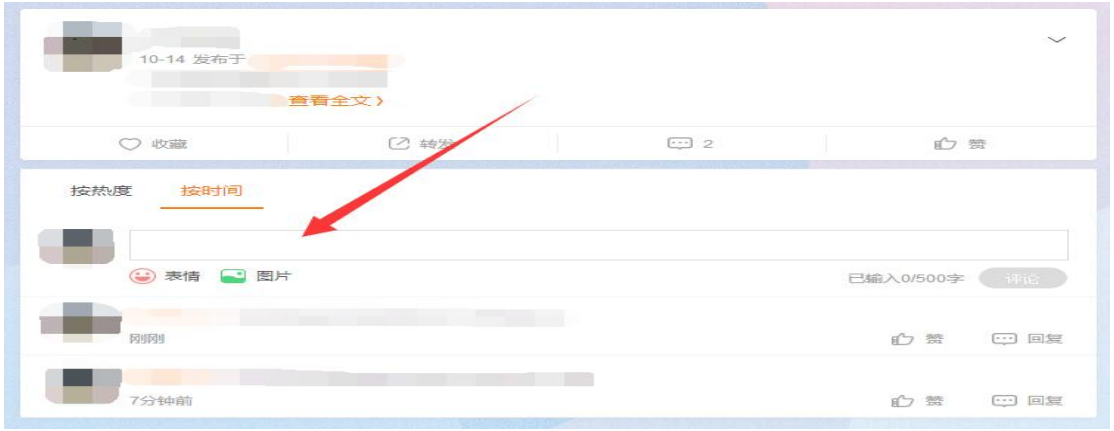
script拦截

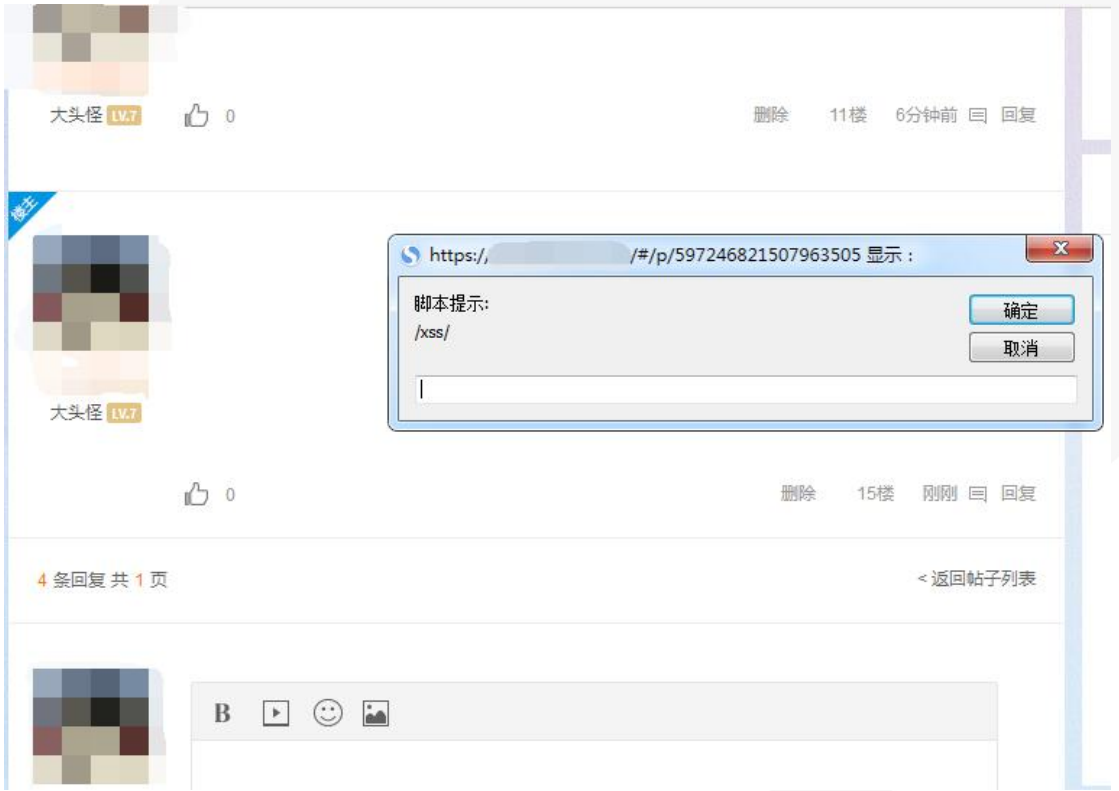
s cript不拦截

再次造成存储型XSS。

安全实例：

考虑其他输入点，尝试构造其他编码，再次突破WAF：





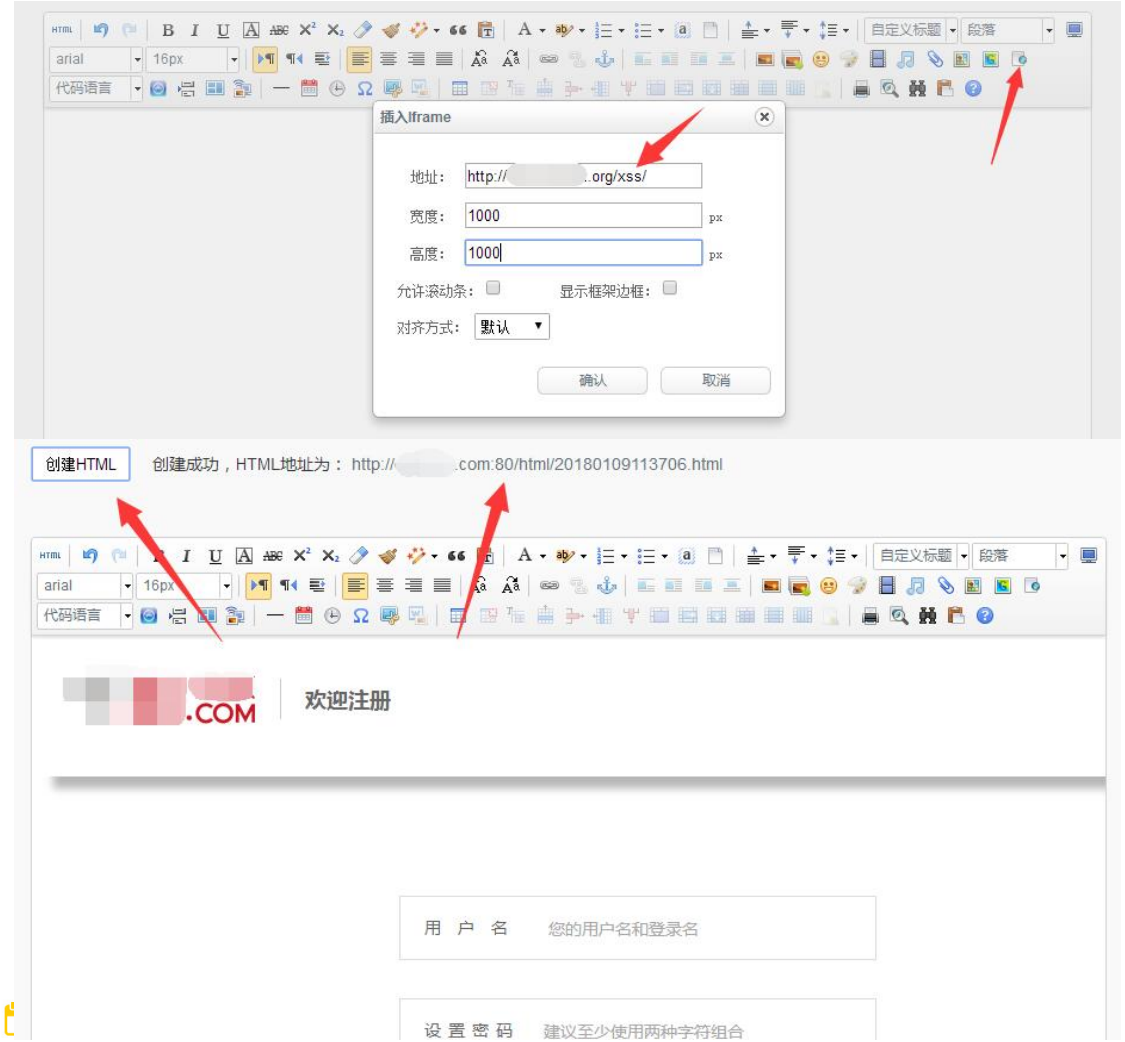
安全实例：

更改名称处：

清空token参数，可CSRF　　designname参数放入poc，该包可无视名称一次设置规律，无限次数更改。（绕过规定的一次）

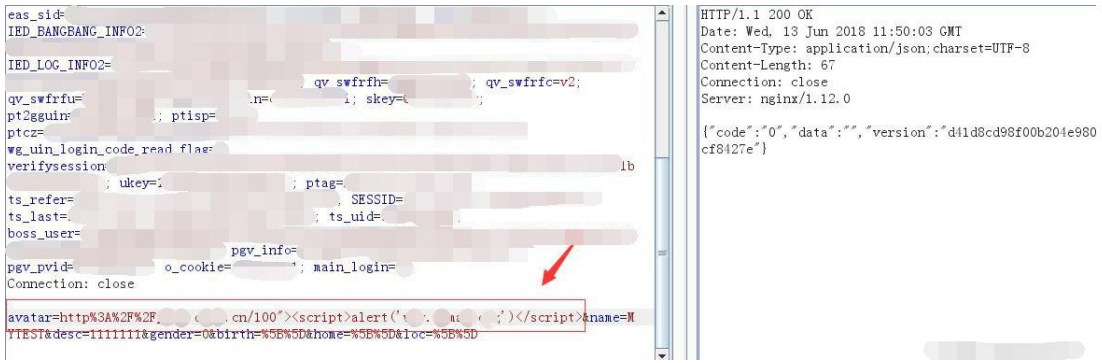
安全实例：

再次修复后，尝试寻找编辑器漏洞：

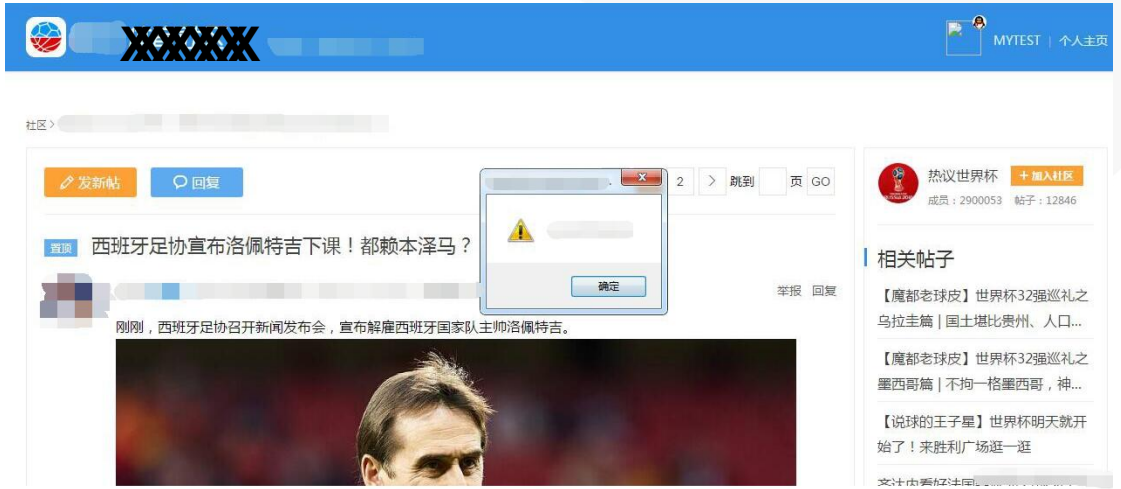


安全实例：

修复后，再次寻找输入点，发现保存资料处，可自定义头像链接：



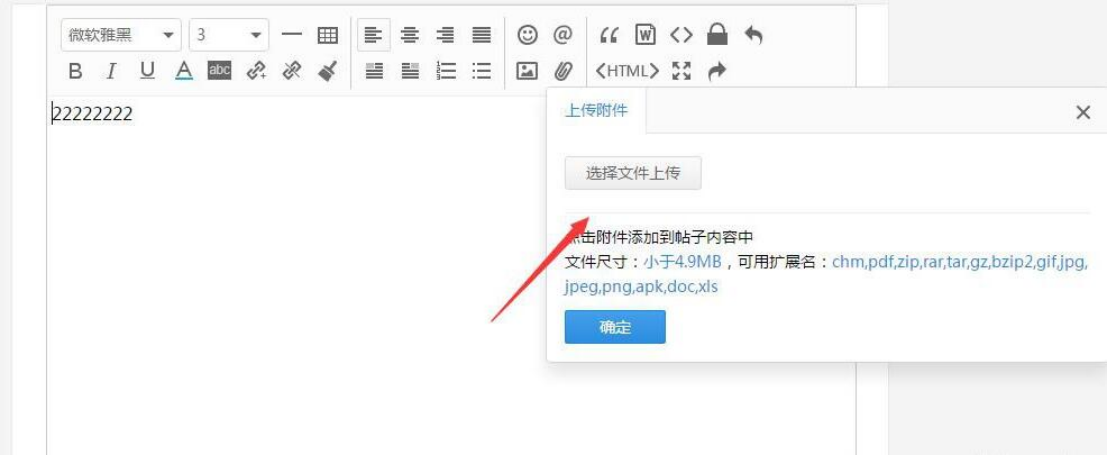
在论坛版块触发：

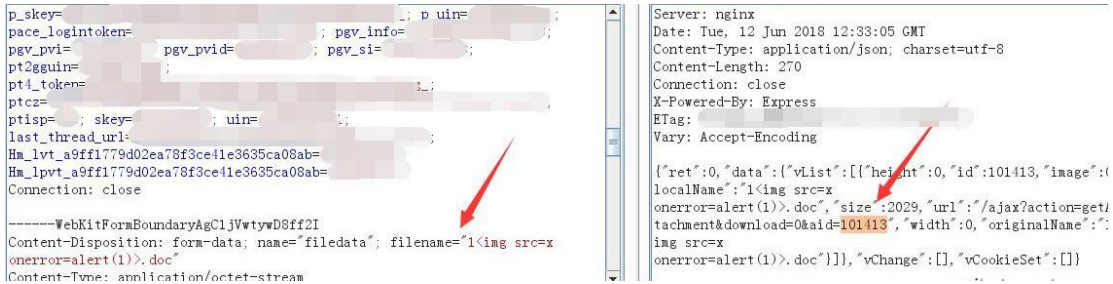


XSS漏洞挖掘与安全防护

安全实例：

修复后再次寻找输入点：
上传附件是一个很好的输入点，且文件名可控：





查看评论触发：

安全实例：

用户的任何输入都是不可信的，XSS防不胜防。

篇幅有限，案例暂时分享这么多。

相关技术：

自动化扫描技术 ：XSS扫描器（只能发现一些简单存在的反射型XSS，比较深入的存储型XSS很难扫描到。）

机器识别技术：针对用户的输入，制定黑名单机制，触发黑名单内标签，进行标记用户并跟踪操作。（https://sec.ly.com/xsspt.txt）

黑产数据鉴别溯源技术：从黑产交易中溯源到问题点。

挖矿存储型捕获技术：根据追踪一些分成挖矿平台的script标签进行溯源。

关键字预警技术：通过论坛内自带搜索功能，或谷歌关键字搜索不应该存在的标签。

安全防护及风险规避：

永远不要相信用户的输入以及潜在输入！

1.HttpOnly防止劫取Cookie
HttpOnly最早由微软提出，至今已经成为一个标准。浏览器会禁止页面的javascript访问带有HttpOnly属性的Cookie。目前主流浏览器都支持，HttpOnly解决是XSS后的Cookie支持攻击。

2.输入检查
一般是检查用户输入的数据中是否包含一些特殊字符，如<、>、'、"等，如果发现存在特殊字符，则将这些字符过滤或者编码。

3.输出检查
通常是输出防御编码，但后面如果是输出到事件或脚本，则要再做一次javascriptEncode编码，如果是输出到HTML内容或属性，则要做一次HTMLEncode。

4.富文体
设置好白名单，严格控制标签。

测试时尽量不影响正常用户使用，不影响正常网站服务，向企业反馈漏洞尽量说清楚问题，不得在未授权下公开漏洞，在SRC提交漏洞一定要阅读提交漏洞须知。

XSS漏洞挖掘与安全防护

THANKS!

MIMAZ.ORG