Tutorials Browser

- Web Pentest
  - Information-Gathering
    - Part-0-Purpose-of-Information-Gathering.html
    - Part-1-information-Gathering-with-websites.html
    - Part-2-information-Gathering-with-Google.html
    - Part-3-information-Gathering-with-nmap.html
    - Part-4-DNS-information-Gathering-with-DIG.html

XSS Series by Securityidiots

Table of content:

9. Misc XSS Techniques(shorter payloads)
10. Stealing CSRF Token and Performing CSRF Actions using XSS
11. Exploiting Self XSSs via Login/Logout CSRF Chain.
12. AngularJs Template Injections to XSS
13. Manual VS Automated Scanning and Tools/methods for XSS testing

BASICS of JAVACSRIPT Part 1 for XSS

In This Tutorial, We are going to discuss the basics of Javascript(js) for this upcoming series of XSS to understand and exploit XSS, way better than just firing alerts.

INTRODUCTION

Javascript is a client-side Programming/Scripting Language for the Web used within HTML for Changing the behavior/functionality of a web page ie. javascript could be used to Change the Style(CSS) and Document Object Model(DOM) of a Web Page and may or may not depend on the user interaction/input.

What is DOM(Document Object Model)?

Simply put, When a Web Page is loaded a DOM Tree(Data Structure) is Created Which contains all the HTML Nodes including Elements/Tag Names, attributes of the tags, id of the tag, name of the tag, text/html content within another HTML element ie. DOM contains the structure of whole Web page.

A Simple HTML5 Page would be like This:

```
<!doctype html>
<html>
<body>
Some Text
<p name="para" id="1">Hi</p>
</body>
</html>
```

DOM would look like this:

```
#document
|_DOCTYPE: html
|_HTML
    |_HEAD
    |_BODY
      |_#text: Some Text
      |_P name="para" id="1"
      |      #text: Hi
      |_#text:
```

Different ways to execute JS:

- Inside <script>...</script>
- Loading External Js file with 'src' attribute
- Using Event handlers
- Browser console using F12

Javascript Comments:

- Single Line Comment : //This is a Comment
- Another Single Line comment <!--This is commented
- Multi Line Comment : /*Multi line comment*/
- Lesser Known Single Line Comment : --> This is also a Comment

DataTypes & Variables declaration in JS

The Syntax is bit Similar to C ie. each line ends with semicolons (they are not required when statements are written in different lines)etc.. except that we don't need to declare data types of variables for example:

```
var a=10,b=null,c="String",d=false,e={A:1,B:"String2",C:[1,2]},f=[1,2,3,"String3"],g; //Explained Below
```

We can also declare the same in different lines

```
var a=10;
var b=-20;
```

a is an integer data type
b is an Empty Object
c is a String
d is a boolean (true/false) data type
e is a javascript Object

Basic Syntax in javascript to declare objects, since javascript objects are declared within braces as {} returns an Empty Object. An object is an array of multiple properties, value of a property could be any datatype. Check a few example objects below

```
e={propertyname:"value of property"}
eg: {propertyname1:"value1",propertyname2:"value2"}
```

Try and define the below variables using any of the method mentioned above and then check their values, You can simply run this in Browser Console(Press F12 on Chrome/FF), paste the below code and then type in thier name to check each variable's value one by one.

```
var a=10,b=null,c="String",d=false,e={A:1,B:"String2",C:[1,2]},f=[1,2,3,"String3"],g;
```

An Example HTML Document for the above variables would be :

```
<!doctype html>
<html>
<script>
var a=10,b=null,c="String",d=false,e={A:1,B:"String2",C:[1,2]},f=[1,2,3,"String3"],g;
</script>
</html>
```

Let us see how to access Javascript Objects, first of all using the below statement in javascript console we will declare an object "obj" which have two properties prop1 and prop2, using the below statement.

```
var obj={prop1:300,prop2:"String"};
```

The values of These Properties Could be accessed in 2 Ways

1. Dot Notation

```
obj.prop1; // would return the value of property prop1
obj.prop2; // would return the value of property prop2
```

2.Square Brackets [] Notation

```
obj["prop1"];// would return the value of property prop1
obj["prop2"];// would return the value of property prop2
```

Window and Document Object in javascript:

Window is the main JavaScript global object, accessed by "window" object it contains all the information about the opened window (height, width, name of the window, DOM etc) and also its opener window (if its there). Document is a property of "window" object only but since "window" object is global it could be accessed by "document" or even as "window.document".

So basically, DOM (Document Object Model) loads all the objects inside "WINDOW" and then the "DOCUMENT" object gets loaded inside the window object. Some of the Important Properties of "window" & "document" objects are:

1. **Window.location** or location;, which contains all the information about the current loaded Window like the URL, Hostname, port, protocol, path etc.

```
window.location.host : "www.host.com"
window.location.hostname  : "www.hostname.com"
```

```
window.location.href  : "https://www.hostname.com/path/to/file.php"
window.location.origin  : "https://www.hostname.com"
window.location.pathname  :"/path/to/file.php"
window.location.port :""
window.location.protocol  : "https:"
```

Changing the properties of location Object would redirect the page for example location.href="https://google.com"; would redirect to https://google.com

2. **Window.opener**; returns the reference to the parent window which opened the current window using window.open() method for example:

```
window.open( URL, name); //Opens a Window of name and URL , if url is empty string it returns about:blank

mywin=window.open("", "MyWindow" );//opens an about:blank window with name "MyWindow"

mywin.document.write("This is Child Window");

mywin.opener.document.write("The Parent Window");//We Could Change content of the Parent Window
```

Author : Rahul Maini
Date : 2017-04-22