Computer Security – Lab 3
Cross Site Scripting

Thomas Kerber

November 12, 2016

1 XSS Game

For this lab, you should play through a google training game for cross site scripting (XSS) attacks, found at `https://xss-game.appspot.com/`. The game consists of six levels, each exploiting different vulnerabilities in toy websites. **Level 3 is broken in Firefox. The game was designed for, and works in, Google Chrome. Please use Chrome for this lab.**

2 Basic concepts of HTML and JavaScript

This lab requires some basic understanding of HTML and JavaScript. If you already have this, great. If you do not, this section will give you the most basic concepts you need to know to complete the lab. It is not the aim of this course, or this lab to teach you either HTML or JavaScript, and for a proper introduction you may find `http://w3schools.com` a helpful resource.

2.1 HTML

HTML is a markup language used to describe web pages. Broadly speaking, it consists of **tags** and text. Tags are indicated by angular braces, e.g. `<div>`. Tags can have a matching closing tag, indicated by a forward slash after the opening brace: `</div>`. Tags can also "close themselves", by ending with a forward slash before the closing brace: `<img />`.

Tags can also have attributes, which have a **name** and a **value**. Different tags have different valid attribute names, each with an associated meaning for the value. Attributes are specified inside a tag following the syntax `<tag attribute="value">`. Single quotes are also permissible: `<tag attribute='value'>`.

HTML is also **fault tolerant**. If the HTML given to a browser is not entirely correct, it will take a guess and evaluate it anyway. E.g. `<p>foo<p>bar` will be interpreted as `<p>foo</p><p>bar</p>`.

2.2 JavaScript

The basic syntax of JavaScript required for this lab should be familiar to you. A function '`foo`' is called with `foo()`. It can be passed arguments if desired: `foo(bar)`, `foo(bar, baz)`, etc. Strings are delimited by either single, or double quotes, i.e. `'foo'` and `"foo"` define the same string. Strings can be concatenated with the `+` operator, e.g. `"foo" + 'bar'` is equal to `"foobar"`. Separate statements can be placed on separate lines, and/or separated by a semicolon.

2.3 JavaScript in HTML

Finally, a few details on invoking JavaScript from HTML. First, the `<script>` tag is used to include JavaScript directly. Everything between the opening and closing script tags will be executed as JavaScript. E.g. `<script>foo()</script>` will call the function `foo`. Next, many tags support attributes which can include JavaScript. These include `onmouseover`, `onclick`, and `onerror`. A tag which supports this is, for example, the `<img>` tag, which is used to load images (The image URL is specified through the `src` attribute). Finally, JavaScript can be executed by navigating to a URL beginning with ‘`javascript:`’. In this case, the browser will execute the remainder of the URL as JavaScript. A word of caution though: Just navigating to ‘`javascript:alert()`‘ will not have the desired effect for this lab, as it will not execute the JavaScript in the context of the vulnerable website.