Alexander Pearsall

Gavin Danner-Rivers

Lab: Tweeter

Section 121

# Lab Tweeter

## Security Research

**Research cross site scripting attacks. Describe cross site scripting in your own words.**

Cross-site scripting (XSS) attacks are a type of attack that allows the ability for malicious scripts to be ran on an unsecure website. These types of attacks are most commonly present where a web application accepts user input, such as text and input fields. With these text fields, attackers have the ability to inject HTML and JavaScript code inside it, which will ultimately run the inserted text which has the chance of causing damage to the website.

**Attack mechanisms are sometimes called "vectors". From your research, what are two different vectors that attackers use to inject JavaScript in a web application?**

The two main vectors that attackers use are Cross Site Scripting (XSS) and Cross-Site Request Forgery (CSRF).

**What strategies can a web application developer use to prevent these attacks?**

In order to prevent any attacks from being done on a website, developers should ensure that attackers cannot allow cross-site scripting or cross-site request forgery on their website. The first option someone can take to prevent any XSS on their website is to validate any input from the user. Ensure that the characters being entered are what is expected. If the input only requires numerical values, ensure that those values can only be inputted. Another security option would be to ensure that is being outputted is does not modify the contents of the website as well. Checking to make sure the HTML, JavaScript, or other contents were not modified ensures that none of the code was changed for the worse.

## Local Storage Research

## Local Storage

* Local storage will maintain its data indefinity, even if the browser is closed and reopened.
* Specific data in local storage can be accessed across multiple browser tabs and windows.
* Local storage will not expire after a certain period of time
* Local storage can only be removed with JavaScript, by clearing the browser's cache, or by clearing the locally stored data.

**Session Storage**

* Session storage will store data for as long as the browser is in session. The browser will keep its session storage even when the page reloads. If the browser or its tab has been deleted, the session storage will be deleted.
* Specific data in session storage can only be accessed by a single browser tab or window.
* Session storage data is never transferred to a server.
* The storage of a session cannot exceed the size of a cookie (around 5mb).

**Why are there two different mechanisms? Give a specific example of why you'd use one vs the other.**

Session storage and local storage can be utilized for different purposes. For example, you’d want to use local storage to store any user preferences or data that can be accessed while offline. You would utilize session storage when it comes to storing something as an item in a shopping cart or text typed out in a text field.

**Suggestions and/or Notes**

We enjoyed using making the application and using local storage to store data.

## Sources

<https://owasp.org/www-community/attacks/xss/>

<https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html><https://www.geeksforgeeks.org/cross-site-scripting-xss-prevention-techniques/><https://developer.mozilla.org/en-US/docs/Web/API/Web_Storage_API>

<https://www.geeksforgeeks.org/difference-between-local-storage-session-storage-and-cookies/>