MS-ISAC

MS-ISAC Security Primer

Cross-Site Scripting (XSS)

March 2017, SP2017- 0615

Cross-site scripting (XSS) vulnerabilities allow a malicious cyber actor to insert and execute unauthorized code in a web application. A web application is a component of a website that allows for a function to be performed from a user's web browser, for example viewing email or managing a shopping cart. Successful XSS attacks can be used to execute malicious code, display unauthorized images or text, obtain session details, and/or provide the attacker unauthorized access to confidential information.

There are two main types of XSS vulnerabilities: reflected and stored. Reflected, or non-persistent, XSS vulnerabilities are more common and occur when a web application returns dynamic content based on user-entered data, such as during the search query and result process. Reflected vulnerabilities are the result of insufficient sanitization of the user request. Stored or persistent XSS vulnerabilities occur when user-entered data is stored on a webserver, such as when a user makes a post to an online message board. This vulnerability is often the result of improper HTML escaping.[1] Stored XSS attacks are generally more severe and result in more compromised users and/or data than reflected attacks.

An attacker can discover if a website is vulnerable to XSS by testing a web application, such as a login page or search function. The attacker will attempt to run a simple script by inputting a basic HTML command, such as displaying a pop-up window. If the window appears, it indicates the input is not being checked and the website is vulnerable to XSS. Once it is determined that a web application is vulnerable, an attacker may distribute a link to the affected website or hide HTML code in the user-entered data. By distributing a link with the additional malicious code hidden in the link, the attacker is attempting a reflected XSS attack. When the attacker hides HTML code in the user-entered database, the attacker ensures that every time the user-entered data is shown, the code is also executed.

TECHNICAL RECOMMENDATIONS:

- Conduct a vulnerability scan of Internet-facing applications, focusing on identifying and remediating XSS vulnerabilities and patching out-of-date software, especially content management systems.
- If your website is hosted by a third party, establish a relationship with your hosting provider and have its contact information readily accessible in case of a compromise. Ensure that the hosting provider conducts regular vulnerability scans and updates the website to address vulnerabilities.
- Sanitize user input, including the HTTP referrer objects, GET and POST parameters, URLs, and form data, prior to executing or storing it
- Encode output and filter input for HTML special characters. Only accept expected user input and limit input length.

The MS-ISAC is the focal point for cyber threat prevention, protection, response, and recovery for the nation's state, local, tribal, and territorial (SLTT) governments. More information about this topic, as well as 24x7 cybersecurity assistance for SLTT governments, is available at 866-787-4722, SOC@cisecurity.org, or https://msisac.cisecurity.org/. The MS-ISAC is interested in your comments - an anonymous feedback survey is available at: https://www.surveymonkey.com/r/MSISACProductEvaluation.

[1] HTML escaping allows characters that indicate a programming reference in HTML to instead be interpreted as general characters. For instance, angle brackets (“”) enclose HTML tags, which are interpreted by web browsers as code. By escaping the angle brackets, the brackets are interpreted as characters (represented as “<” and “>”), which prevent the tag from executing.