A7-Cross-Site Scripting (XSS)

Automated tools can detect and exploit all the forms of XSS. The three forms of XSS are Reflected XSS, Stored XSS and DOM XSS. Reflected XSS is the application or API includes the user input as a part of HTML output. The attacker can have a successful attacks when the attacker executed arbitrary HTML and JavaScript in the browser of the victim. The next form is the Stored XSS, this is the application stores the user who does not have privileges input that is viewed at a later time by another user or the administrator. This form is often considered a critical risk. The last form of XSS is DOM. DOM XSS is when the Java scripts frameworks, single page application and API that include attacker controllable data to a page are vulnerable to DOM XSS.

Medication

To prevent XSS, applying context sensitive encoding when modifying the browser document on the client side acts against DOM XSS. Similar context sensitive escaping techniques can be applied to browser API’s when this cannot be avoided. Another solution is to escape untrusted HTTP request because it will resolve Reflected and Stored XSS vulnerabilities. The developer can use frameworks that automatically escape XSS by design.

Example

One example of this threat is when the application uses untrusted data in the construction of the HTML snippet without validation. The attack causes the session ID of the victim to be sent on the attacker’s website. This will allow the attacker to hijack the current session of the victim.