eJPT Certification Section

Section: Cross Site Scripting

09/18/20

Why is it important?

* Ability to attack web applications’ users
* Ability to control web applications content
* Gain advanced web attacks skills

**Cross Site Scripting**

* XSS is a vulnerability that lets an attacker control some of the content of a web application.
  + Allows attacker to target web app users.
* XSS enables attackers to:
  + Modify the content of the site at run-time.
  + Inject malicious contents;
  + Steal the cookies, thus the session, of a user.
  + Perform actions on the web application as if it was a legitimate user
  + and there are many other benefits of xss.
* Three actors involved in an XSS attack are:
  + The web app
  + Web app user
  + The pentester
* XSS vulnerabilities happen when a web app uses unfiltered user input to build the output content displayed to its end users; this allows the attacker to control the output of the HTML and JS code.
  + User input is any parameter coming from the client side of the web app such as:
    - Request headers
    - Cookies
    - Form inputs
    - POST parameters
    - Get parameters
  + All of these input channels should be validated by server side security functions.
  + XSS involves injecting malicious code into the output of a webpage. The malicious code is then rendered or executed by the browser of the visiting users.
* **Attackers**
  + Malicious users exploit xss vulnerabilities to attack the users of a web site by:
    - Making their browsers load malicious content
    - Performing operations on their behalf, like buying a product or changing a password.
    - Stealing their session cookies, thus being able to impersonate them on the vulnerable site.
* **Finding an XSS**
  + To exploit an XSS vulnerability that you find, you need to know the type of cross-site scripting attack you are carrying out. Cross-site scripting vulnerabilities can be reflected, persistent, or DOM based.
  + **Reflected**
    - This is when a malicious payload is carried inside the request that the browser of the victim sends to the vulnerable website.
    - Could be triggered by posting a link on a social network or via phishing campaign.
      * When the user clicks on the link, they trigger the attack

**A screenshot of a cell phone

Description automatically generated**

* + **This type of attack is called reflected because an input field of the HTTP request sent by the browser gets immediately reflected to the output page.**

**A screenshot of a cell phone

Description automatically generated**

* There are more advanced XSS reflected attacks that can bypass anti xss-filters but they are out of the scope of this certification.
* However, persistent xss attacks can bypass these filters.
* **Persistent XSS Attacks**
  + This is when the xss payload is sent to the vulnerable web server and then stored.
    - Then, when the web page of the vulnerable web app pulls the stored malicious code and then puts it within the HTML output, it will deliver the XSS payload.
    - This is called a persistent xss attack because the malicious code gets delivered each and every time a web browser hits the “injected” web page.
  + The most common vector for persistent attacks are HTML forms that submit content to the web server and then display that content back to the users.
    - Elements such as comments, user profiles, and forum posts are potential vector for XSS attacks.
  + **A screenshot of a cell phone

    Description automatically generatedPersistent XSS Attacks Example**
* **Cookie Stealing via XSS**
  + JS can access cookies if they do not have HttpOnly flag enabled; this means that an xss attack can be used to steal the cookies. In many cases if an attacker can get a hold of your cookie that means they can steal your session and impersonate you.

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**