Mitigating Cross Site Scripting Attacks

The following sections describe the impact of the XSS attack and the approaches you can use to mitigate it. **Note** that XSS attacks are prevented on the latest WSO2 products by default. This is due to output encoding of the displaying values. However, if additional protection is required, an input validation valve can be configured as explained below.

- How can XSS attacks be harmful?
- Mitigating XSS attacks

### How can XSS attacks be harmful?

Cross Site Scripting (XSS) attacks use web applications to inject malicious scripts or a malicious payload, generally in the form of a client side script, into trusted legitimate web applications. XSS Attackers can gain elevated access privileges to sensitive page content, session cookies, and a variety of other information with respect to web applications that are maintained by the web browser on behalf of the user.

### Mitigating XSS attacks

You can use the following approach to mitigate XSS attacks.

#### Mitigating using the XSS Valve

The XSS Valve acts as a filter to differentiate between the malicious scripts from the legitimate scripts by carrying out a specific validation on the URL patterns.

#### Configuring the XSS Valve

1. Open the `<PRODUCT_HOME>/repository/conf/carbon.xml` file and add the following code snippet under the `<Security>` tag.

   ```xml
   <XSSPreventionConfig>
     <Enabled>true</Enabled>
     <Rule>allow</Rule>
     <Patterns>
       <Pattern>commonauth</Pattern>
     </Patterns>
   </XSSPreventionConfig>
   ```

2. Add the following configuration within the `<Hosts>` element of the `<PRODUCT_HOME>/repository/conf/tomcat/catalina-server.xml` file.

   ```xml
   <Valve className="org.wso2.carbon.ui.valve.XSSValve"/>
   ```

3. Restart the product server.