

InputValidation_ReflectedXSS_Edit_Finding - Findings Report

Oct 26, 2007 7:07:24 PM

Classification	Type	Severity	File	Line	CWE ID	SmartTrace
Vulnerability	Validation.Required		Z:\jspwiki\JSPWiki_2_4_104\JSPWiki-src\web-root\JSPWiki.war\Edit.jsp	92	20	
<p>Description: The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.</p> <p>Exmaple 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp</p> <p>Recommendation: Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.</p> <pre>// Someone changed the page while we were editing it! // log.info("Page changed, warning user."); session.setAttribute(EditorManager.REQ_EDITEDTEXT, EditorManager.getEditedText(pageContext)); response.sendRedirect(wiki.getUrl(WikiContext.CONFLICT, pagereq, null, false)); return; } //</pre>						
Vulnerability	Validation.Required		Z:\jspwiki\JSPWiki_2_4_104\JSPWiki-src\web-root\JSPWiki.war\Comment.jsp	75	20	
<p>Description: The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.</p> <p>Exmaple 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp</p> <p>Recommendation: Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.</p> <pre>{ link = HttpUtil.retrieveCookieValue(request, "link"); if(link == null) link = ""; } session.setAttribute("link", link);</pre>						

```
//
// Branch
//
log.debug("preview="+preview+", ok="+ok);
```

Vulnerability

Validation.Required



Z:\jspwiki\JSPWiki_2_4_104\JSPWiki-src\web-root\JSPWiki.war\Edit.jsp

169

20



Description:

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Exmample 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
return;
}
else if( preview != null )
{
    log.debug("Previewing "+pagereq);
    session.setAttribute(EditorManager.REQ_EDITEDTEXT,
        EditorManager.getEditedText(pageContext));
    session.setAttribute("author",user);
    session.setAttribute("link",link != null ? link : "" );

    session.setAttribute("changenote", changenote != null ? changenote : "" );
```

Info

Info



Z:\jspwiki\JSPWiki_2_4_104\JSPWiki-src\web-root\JSPWiki.war\Edit.jsp

169

Description:

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Exmample 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
return;
}
else if( preview != null )
{
    log.debug("Previewing "+pagereq);
    session.setAttribute(EditorManager.REQ_EDITEDTEXT,
        EditorManager.getEditedText(pageContext));
    session.setAttribute("author",user);
    session.setAttribute("link",link != null ? link : "" );

    session.setAttribute("changenote", changenote != null ? changenote : "" );
```

**Description:**

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Exmaple 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
else if( preview != null )
{
    log.debug("Previewing "+pagereq);
    session.setAttribute(EditorManager.REQ_EDITEDTEXT,
        EditorManager.getEditedText(pageContext));
    session.setAttribute("author",user);
    session.setAttribute("link",link != null ? link : "" );

    session.setAttribute("changenote", changenote != null ? changenote : "" );
    response.sendRedirect( wiki.getURL(WikiContext.PREVIEW,pagereq,null,false) );
    return;
}
```

**Description:**

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Exmaple 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
// Someone changed the page while we were editing it!
//
log.info("Page changed, warning user.");
session.setAttribute( EditorManager.REQ_EDITEDTEXT, EditorManager.getEditedText(pageContext) );
response.sendRedirect( wiki.getURL(WikiContext.CONFLICT, pagereq, null, false) );
return;
}
//
```

**Description:**

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Exmaple 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
}  
//  
// WYSIWYG editor sends us its greetings  
//  
String htmlText = request.getParameter( "htmlPageText" );  
if( htmlText != null && cancel == null )  
{  
    text = new HtmlStringToWikiTranslator().translate(htmlText,wikiContext);  
}
```

Info

Info



Z:\jspwiki\JSPWiki_2_4_104\JSPWiki-src\web-root\JSPWiki.war\Edit.jsp

171

Description:

The Edit.jsp will use a variety of different request parameters directly without validation and set session attributes with this tainted data. Later in different application components (JSPs) these values will be used directly (sometimes without proper output encoding). It is recommended that these values be properly validated prior to setting them into the session as attributes.

Example 1: link is used as a hidden field from the session attribute directly, which is set in Edit.jsp

Example 2: remember is used as a hidden field here in Edit.jsp, it is set in Comment.jsp

Recommendation:

Validate each parameter prior to setting the value into the session attribute. Output Encode the value rendered to the user. Use the "StringUtil.replaceEntities()" method.

```
else if( preview != null )  
{  
    log.debug("Previewing "+pagereq);  
    session.setAttribute(EditorManager.REQ_EDITEDTEXT,  
        EditorManager.getEditedText(pageContext));  
    session.setAttribute("author",user);  
    session.setAttribute("link",link != null ? link : " ");  
  
    session.setAttribute("changenote", changenote != null ? changenote : " ");  
    response.sendRedirect( wiki.getUrl(WikiContext.PREVIEW,pagereq,null,false) );  
    return;  
}
```