**CHƯƠNG 23: PLUG-INS**

Struts plug-in là một cơ chế cung cấp và hỗ trợ tái sử dụng lại code. Plugin-in cơ bản là một file JAR. Trong đó chứa nhiều Java class, FreeMarker, hoặc Velocity templates, và một file struts-plugin.xml. Sau đây là cách cấu hình ứng dụng sử dụng plug-in.

1. **Tổng quan:**

* Struts được thiết kể có thể mở rộng thêm plug-ins. Cách sử dụng plug-in dễ dàng bằng cách copy file plug-in JAR và thư mục **WEB-INF/lib**. Không giống như một tập tin JAR bình thường, một plug-in có thể có chứa file **struts-plugin.xml** tuân thủ các quy tắc tương tự như file **struts.xml.** File này thiết lập cấu hình trong plug-in, và khi Struts load các file cấu hình theo thứ tự sau:
  + File **struts-default.xml** trong **struts2-core-[VERSION].jar**
  + Tất cả các file **struts-plugin.xml** trong các plug-ins của ứng dụng
  + **File struts.xml**
* Do đó, chúng ta có thể định nghĩa lại các giá trị của file **struts-default.xml** trong file **struts-plugin.xml**, dĩ nhiên file **struts.xml** sẽ là file định nghĩa lại các giá trị cuối cùng của các file khác.
* Chúng ta có thể phân bố lại các thành phần struts trong plug-in, như thêm package, result type, custom interceptors, actions, tag libraries…

1. **The Plug-in Registry**

* Struts có đi kèm một số plug-in, như Tiles plug-in, JfreeChart plug-in, và SiteMesh plug-in. Tuy nhiên, khi xài Struts người ta vẫn thích thú với plug-in bên ngoài khác, hầu hết là miễn phí. Web site tham khảo này có chứa các plug-ins Registry của Struts2:

<https://cwiki.apache.org/S2PLUGINS/home.html>

1. **Writing A Custom Plugin**

* Viết plug-in không khó, nếu bạn biết cách tạo một file JAR, bạn cũng có thể tạo một plug-in. Ở ứng dụng **app23a** có chưa một result type mới là CaptchaResult class (**Chương 19 – “Custom Result Types”)**
* CAPTCHE result type dự vào class **CaptchaResult**, lớp này được extends **StrutsResultSupport**. In order for the result type to be easily used in applications, you need to package it as a plug-in. Ví đây là một result type, bạn cần đăng kí nó trong **struts-plugin.xml**. **Listing 23.1** là file **struts-plugin.xml**.

Listing 23.1 struts-plugin.xml

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <!DOCTYPE struts PUBLIC "  -//Apache Software Foundation//DTD Struts Configuration 2.0//EN"  "http://struts.apache.org/dtds/struts-2.0.dtd">  <struts>  <package name="captcha-default" extends="struts-default">  <result-types>  <result-type name="captcha"  class="com.brainysoftware.captcha.CaptchaResult" />  </result-types>  </package>  </struts> |

* Cấu trúc thự mục trong ứng dụng plug-in của chúng ta như hình 23.1 Gồm 1 class và một file XML

**Hinh 23.1 Cấu trúc thư mục của captcha plug-in**



* Tiếp theo là bước tạo file JAR. Cách chuẩn nhất (không phải dễ nhất) là sử dụng lệnh **jar** (command line) có trong JDK. Điều kiện là JDK của bạn phải được add được dẫn để bạn có thể gọi lệnh jar bất kì nơi nào trong máy tính.
  + 1.Thay đổi thư mục hiện hành đến thư mục có **struts-plugin.xml** (trong môi trường cmd). Thư mục này cũng sẽ chứa thư mục **com**
  + 2. Đánh dòng lệnh sau và Enter

jar -cvf captchaplugin.jar \*

* Một file JAR có tên là captchaplugin.jar sẽ được tao, đây là file JAR mà chúng ta làm plug-in.

1. **Using the Captcha Plug-in**

* Sau đây là ứng dụng **app23b** sẽ minh họa cách sử dụng plug-in Captcha đã thảo luận trước đó. Đầu tiên, bạn phải copy file JAR và thư mục **WEB-INF/lib** của ứng dụng. Ngoài ra, bởi vì các plug-in sử dụng các class trong Brainy Software's CAPTCHA component, bạn cũng phải copy **brainycaptcha.jar.**
* Ứng dụng có 3 action là **Login\_input**, **Login**, và **GetCaptchaImage**. Các action này được khai báo ở **Listing 23.2**

**Listing 23.2 Khai báo action**

|  |
| --- |
| <package name="app23b" extends="captcha-default">  <action name="Login\_input">  <result>/jsp/Login.jsp</result>  </action>  <action name="Login" class="app23b.Login">  <result name="success">/jsp/Thanks.jsp</result>  <result name="input">/jsp/Login.jsp</result>  <param name="hashCookieName">hashCookie</param>  </action>  <action name="GetCaptchaImage">  <result type="captcha">  <param name="hashCookieName">hashCookie</param>  <param name="wordLength">6</param>  <param name="imageWidth">90</param>  <param name="imageHeight">25</param>  </result>  </action>  </package> |

* Đầu tiên, bạn phải chú ý đến thuộc tính **extends** trong thẻ **package**. Nó mang giá trị là **captcha-default**, là đại diện cho một package trong Captcha plug-in. Vì **captcha-default** mở rộng **struts-default**, bạn có thể kế thừa tất cả các thiết lập bên trong package này. Ngoài ra, bạn có thể dùng result type captche mới (action GetCaptchaImage có result type là captcha).
* Sau đây là class Login ở Listing 23.3

**Listing 23.3 Class Login**

|  |
| --- |
| package app23b;  import javax.servlet.http.Cookie;  import javax.servlet.http.HttpServletRequest;  import org.apache.struts2.interceptor.ServletRequestAware;  import com.brainysoftware.captcha.CaptchaUtil;  import com.opensymphony.xwork2.ActionSupport;  public class Login extends ActionSupport  implements ServletRequestAware {  private String userName;  private String password;  private String word;  private String hashCookieName = "hash";  private HttpServletRequest  httpServletRequest;  public void setServletRequest(HttpServletRequest  httpServletRequest) {  this.httpServletRequest = httpServletRequest;  }  // getters and setters not shown  public String execute() {  Cookie[] cookies = httpServletRequest.getCookies();  String hash = null;  for (Cookie cookie : cookies) {  if (cookie.getName().equals(hashCookieName)) {  hash = cookie.getValue(); break;  }  }  if (hash != null  && userName.equals("don")  && password.equals("secret")  && CaptchaUtil.validate(word, hash)) {  return SUCCESS;  }  else {  addActionError("Login failed.");  return INPUT;  }  }  } |

* Chúng ta đặc biệt chú ý phương thức **execute**. Cách thức hoạt động đã được giải thích trong chương 19. Người dùng có thể đăng nhập bằng username là don, password là secret, và nhập từ từ hình ảnh CAPTCHA.
* Trang Login.jsp thể hiện Login form. Chi tiết trang này ở **Listing 23.4**. Trang Thank.jsp là trang bạn thấy sau khi login thành công, ở **Listing 23.5**

**Listing 23.4 Trang Login.jsp**

|  |
| --- |
| <%@ taglib prefix="s" uri="/struts-tags" %>  <html>  <head>  <title>Login with CAPTCHA</title>  <style type="text/css">@import url(css/main.css);</style>  </head>  <body>  <div id="global">  <h3>Enter your user name, password, and the image word</h3>  <s:actionerror/>  <s:form action="Login">  <s:textfield name="userName" label="User Name"/>  <s:password name="password" label="Password"/>  <tr>  <td><img src="GetCaptchaImage.action"/></td>  <td><s:textfield theme="simple" name="word" value=""/></td>  </tr>  <s:submit value="Login"/>  </s:form>  </div>  </body>  </html> |

**Listing 23.5 Trang Thank.jsp**

|  |
| --- |
| <%@ taglib prefix="s" uri="/struts-tags" %>  <html>  <head>  <title>Thank you</title>  <style type="text/css">@import url(css/main.css);</style>  </head>  <body>  <div id="global"> You're logged in. </div>  </body>  </html> |

* Chú ý, bạn phải copy cả 2 file brainycaptcha.jar và captchaplugin.jar vào WEB-INF/lib của project
* Để kiểm tra, bạn vào URL sau:

<http://localhost:8080/app23b/Login_input.action>

* Bạn sẽ thấy hình ảnh captcha trong trang Login ở **hình 23.2**

**Hình 23.2 – CAPTCHA trong trong Login**



1. **Tổng kết:**

Struts cung cấp một cách tái sử dụng code bằng plug-ins. Trong chương này đã chỉ cách viết và sử dụng nó.