RESTful Service with SSO Protection

**RESTful Service Provider:** (dashboard, commonReport, shipmentMonitor, todoList…)

 新增 TC\_USER



新增 TC\_GROUP



新增 TC\_USERGROUP



修改 glassfish-web.xml

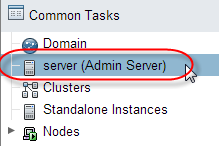
|  |
| --- |
| <security-role-mapping>  <role-name>REST\_USER</role-name>  <group-name>REST\_USER</group-name>  </security-role-mapping> |

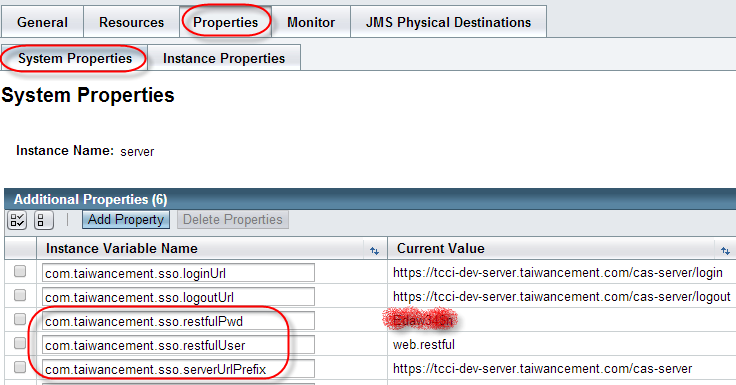
修改 web.xml

|  |
| --- |
| <security-role>  <role-name>REST\_USER</role-name>  </security-role>  …  <security-constraint>  <web-resource-collection>  <web-resource-name>RESTful Service</web-resource-name>  <url-pattern>/service/cementprod/user/\*</url-pattern>  <url-pattern>/service/cementsales/user/\*</url-pattern>  …  </web-resource-collection>  <auth-constraint>  <role-name>REST\_USER</role-name>  <role-name>ADMINISTRATORS</role-name>  </auth-constraint>  </security-constraint> |

**RESTful Service Consumer:** (MyGUI,…)

Server Configuration: (Dev, QA, Prod 已設定完成，本機請參考Dev設定)





Project libraries新增commons-httpclient-3.1.jar, commons-codec-1.7.jar, commons-logging-1.1.1.jar

Project src新增SSOClientFacade.java (copy from MyGUI com.tcci.fc.facade.ssoclient package)

說明: Singleton EJB, 每1小時重新取得web.restful的TGT

修改呼叫RESTful service地方: sample如下(MyGUI生產即時資訊平台 controller)

CementProdContent.java

@EJB

private SSOClientFacade ssoClient;

…

private void executeService() {

/\* restful url pattern

[http://192.168.203.81/dashboard/service/cementprod/user/{loginAccount}/yesterday](http://192.168.203.81/dashboard/service/cementprod/user/%7bloginAccount%7d/yesterday)

\*/

StringBuilder sb = new StringBuilder(dashboardURL);

sb.append("/service/cementprod/user/")

.append(userSession.getAdaccount())

.append('/')

.append(date);

String json = ssoClient.executeSerivce(sb.toString());

Gson gson = new Gson();

vo = gson.fromJson(json, CementProdVO.class);

}