

30c54ef16c ▾

...

emissary / src / main / java / emissary / server / mvc / internal / RegisterPeerAction.java / &lt;&gt; Jump to ▾

ctubbsii Apply changes from formatter plugin update

History

2 contributors

73 lines (63 sloc) | 3.11 KB

...

```
1 package emissary.server.mvc.internal;
2
3 import static emissary.server.mvc.adapters.DirectoryAdapter.DIRECTORY_NAME;
4 import static emissary.server.mvc.adapters.DirectoryAdapter.TARGET_DIRECTORY;
5
6 import java.util.HashSet;
7 import java.util.Set;
8
9 import javax.ws.rs.Consumes;
10 import javax.ws.rs.FormParam;
11 import javax.ws.rs.POST;
12 import javax.ws.rs.Path;
13 import javax.ws.rs.Produces;
14 import javax.ws.rs.core.MediaType;
15 import javax.ws.rs.core.Response;
16
17 import emissary.directory.DirectoryXmlContainer;
18 import emissary.directory.IRemoteDirectory;
19 import emissary.directory.KeyManipulator;
20 import emissary.log.MDCConstants;
21 import org.apache.commons.lang.StringUtils;
22 import org.slf4j.Logger;
23 import org.slf4j.LoggerFactory;
24 import org.slf4j.MDC;
25
26 @Path("")
27 // context is /emissary, set in EmissaryServer
28 public class RegisterPeerAction {
29     private final Logger logger = LoggerFactory.getLogger(this.getClass());
30
31     /*
32      * <!-- Register a peer with the local directory and return the xml --> <Use-Case source="*"
33      * action="/RegisterPeer.action"> <Work type="Bean" target="emissary.comms.http.worker.LogWorker"/> <Work type="Bean"
34      * target="emissary.comms.http.worker.LookupDirectoryWorker"/> <Work type="Bean"
35      * target="emissary.comms.http.worker.RegisterPeerWorker"/> <Work type="Bean"
36      * target="emissary.comms.http.worker.SetProxyWorker"/> <View status="0" view="/xmlDir.jsp"/> <View status="-1"
37      * view="/error.jsp"/> </Use-Case>
38      */
39
40     @POST
41     @Path("/RegisterPeer.action")
42     @Consumes(MediaType.APPLICATION_FORM_URLENCODED)
43     @Produces(MediaType.APPLICATION_XML)
44     public Response registerPeerPost(@FormParam(DIRECTORY_NAME) String directoryName, @FormParam(TARGET_DIRECTORY) String targetDirectory) {
45         if (StringUtils.isBlank(directoryName) || StringUtils.isBlank(targetDirectory)) {
46             return Response.serverError()
47                 .entity("Bad Params: " + DIRECTORY_NAME + " - " + directoryName + ", " + TARGET_DIRECTORY + " - " + targetDirectory).build();
48         }
49         return processRegisterPeer(directoryName, targetDirectory);
50     }
51
52     private Response processRegisterPeer(String peerKey, String dirName) {
53         final IRemoteDirectory dir = new IRemoteDirectory.Lookup().getLocalDirectory(dirName);
54         if (dir == null) {
55             // If we get here, there was a problem looking up the IRemoteDirectory
56             // old RegisterPlaceWorker failure
57             // return new WorkerStatus(WorkerStatus.FAILURE, "Could not register peer " + peerKey);
58             return Response.serverError().entity("Remote directory lookup failed for dirName: " + dirName).build();
59         }
60
61         final Set<String> set = new HashSet<String>();
62         set.add(KeyManipulator.getDefaultDirectoryKey(peerKey));
63         MDC.put(MDCConstants.SERVICE_LOCATION, KeyManipulator.getServiceLocation(dir.getKey()));
64         try {
65             dir.irdAddPeerDirectories(set);
66         } finally {
67             MDC.remove(MDCConstants.SERVICE_LOCATION);
68         }
69
70         logger.debug("Registered peer {} with local dir", peerKey);
71         return Response.ok().entity(DirectoryXmlContainer.toXmlString(dir)).build();
72     }
73 }
```