

[OpenAz] Version 90 - resource-based query and test-resource-directory added

Rich.Levinson [rich.levinson at oracle.com](mailto:rich.levinson@oracle.com)

Tue May 11 01:35:31 EDT 2010

- Next message: [\[OpenAz\] OpenAz Conference Call - Thursday May 13, 1 PM ET](#)
- Messages sorted by: [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

To OpenAz maillist:

Version 90 has been uploaded to the OpenAz project.

This version implements the query capabilities based on an auxiliary OpenAzResourceDirectory, which can be set up to contain multiple resource instances of multiple resource types. The way the initial version works is like this:

- * A query with a scope equal to one of the resource-types will return a Set of all instances of resource-action pairs available for that resource type
- * That set of resource-action pairs will be submitted in a multiple AzResourceActionAssociation AzRequestContext to the sunxacml pdp.
- * The front end SimpleConcreteSunXacmlService will submit each of the Resource-Action pairs, one at a time and collect the results in an AzResponseContext.

In order to implement a reasonable capability, it proved necessary to implement a test resource repository, which is done with the following modules:

- * org.openliberty.openaz.pdp.resources.OpenAzResourceDirectory.java: an interface
- * org.openliberty.openaz.pdp.resources.OpenAzResourceQueryBuilder.java: an implementation of the above interface that takes a scope and AzRequestContext and returns the AzRequestContext populated with the resource-action pairs found in the scope
- * org.openliberty.openaz.pdp.resources.OpenAzTestResourceCollection.java: a "configurable" test collection of resource types and "configurable" resource action pair instances associated with each type, which should allow arbitrary collections of arbitrary resource types to be available for testing.

This complements the capabilities that were demoed in version 88, which will be referred to as "policy-based" query, since that capability is implemented purely with a specific policy dialect designed to return the applicable scopes to the caller. That capability will be put "on hold" for the time being as described below, until it is determined whether it will be operationally useful.

The plan moving forward for "query" is to allow a period of consideration of what capabilities should be added to the initial resource-based query.

In particular, it has become apparent that there is a fairly tight relationship between the syntax that is used to select scopes of resource action pairs from a resource repository and the syntax used for scoping resources in the policies. Also, as resource repositories become large, it will probably be undesirable to do broad scopes on the resources, which will generate large numbers of irrelevant decision requests.

Therefore, strategies are probably desirable to be considered in

order to provide efficient query capabilities, and once those strategies are apparent the build-out of the code should probably be straight-forward.

The next capability to be added is the attribute finder callback, which should be ready to evaluate later this week.

Thanks,
Rich

----- next part -----

An HTML attachment was scrubbed...

URL: <http://lists.openliberty.org/pipermail/openaz/attachments/20100511/7f18876d/attachment.html>

-
- Next message: [\[OpenAz\] OpenAz Conference Call - Thursday May 13, 1 PM ET](#)
 - **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

[More information about the OpenAz mailing list](#)