

Authorizers

by Andreas Hartmann

1. Introduction

An authorizer checks if an identity is authorized to invoke a certain request. The `DelegatingAuthorizerAction` tries to resolve an `AccessController` for the URL. If an `AccessController` could be resolved, its `Authorizer` is used to authorize the request. If no `AccessController` could be found, the access to the request is granted for free.

2. PolicyAuthorizer

A `PolicyAuthorizer` uses policies for authorizing.

3. UsecaseAuthorizer

This authorizer looks for the `lenya.usecase` request parameter and checks the usecase policy file for the roles that are allowed to execute this usecase. The location of this file is defined using the `<config-file>` tag:

```
<authorizer type="usecase">
  <config-file src="context:///lenya/pubs/default/config/ac/usecase-policies.xml"/>
</authorizer>
```

The usecase policy file might look as follows:

```
<?xml version="1.0"?>
<usecases xmlns="http://apache.org/cocoon/lenya/ac/1.0">
  <usecase id="create">
    <role id="editor"/>
  </usecase>
</usecases>
```

4. WorkflowAuthorizer

The `WorkflowAuthorizer` is responsible for protecting workflow transitions. Therefore it

- looks for the `lenya.event` request parameter,
- determines the current state of the workflow instance, and
- checks if the event may be invoked by one of the current roles in this state.