Octopus Framework - Technical

Rudy De Busscher

Version 0.4, ??/??/2018

Table of Contents

Violation handling

Filters

When a violation occurs (user has not the required permission, role, …) the method *isAccessAllowed()* just returns false. This causes the filter chain to abort and thus the real logic (page, rest endpoint, …) is never executed.

Information about which violation occurred is placed as attribute on the servlet request.

name : **octopus.violation.message**, value : **String**

This information is used by the **AccessDeniedHandler** to return an appropriate message (On JSF page, As JSON for JAX-RS, …)

Annotations

When an interceptor handling the annotation detects a violation, it throws an exception **SecurityAuthorizationViolationException**.

TODO

Java SE

OfflineToken

With the help of the PBKDF2, the passphrase is turned into a byte array (using the Processor Id and the first disk UUID as salt)

This is byte array can only be reconstructed with the same passphrase and the same machine information.

This **local secret** is base64 encoded.

The class **OfflineToken** is a JWT wich is signed using the *local secret*.

The JWT is tamper proof, except the user can generate other offlineToken since all the code is available within Atbash Octopus.