Multi-Context Broker

IdP version 2.x only!

Note that the remainder of this document describes the MCB login handler for version 2.x of the Shibboleth IdP. It is strongly suggested that you deploy your MCB functionality in IdPv3, using the resources mentioned above. See Orchestrating Multiple Authentication Methods and Contexts - The Multi-Context Broker (MCB) for more information.

In order to support the InCommon Assurance initiative, a new Shibboleth login handler has been developed. This login handler implements logic to match required authentication contexts to the authentication contexts a user has available.

More Documentation and Sample Configurations

More information, including use cases and sample configurations, is available from the Multi-Context Broker page in the InCommon Assurance wiki. Specifications for what the MCB implements can be found in the RFP.

Source Code & Binary Access

The MCB source code and binary distributions are maintained at Github. https://github.com/Internet2/Shibboleth-Multi-Context-Broker

The Multi-Context Broker (MCB) is installed as a normal Shibboleth login handler. Modifications are made to the handler.xml to specify the schema location and to load it. Additional modifications are made to the web.xml file to load the corresponding MCB authentication servlet. Configuration is done using two files. The first file is a Spring configuration file to load various components as beans. This is the same mechanism used by the Shibboleth internal.xml and service.xml files. A current example is shown below:

Spring Bean Configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:util="http://www.springframework.org/schema/util"
     xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans
/spring-beans-2.0.xsd
                         http://www.springframework.org/schema/util http://www.springframework.org/schema/util
/spring-util-2.0.xsd" >


    <!-- This bean represents an authentication submodule -->
    <bean id="mcb.usernamepassword" class="edu.internet2.middleware.assurance.mcb.authn.provider.
JAASLoginSubmodule">
    </bean>
    <!-- This bean represents an authentication submodule -->
    <bean id="mcb.token" class="edu.internet2.middleware.assurance.mcb.authn.provider.TokenLoginSubmodule">
    </bean>
    <!-- This bean is our configuration object representing the custom configuration file -->
    <bean id="mcb.Configuration" class="edu.internet2.middleware.assurance.mcb.authn.provider.MCBConfiguration">
        <constructor-arg value="/opt/shibboleth-idp/conf/multi-context-broker.xml" />
    </bean>
</beans>
```

The second file represents the custom logic to map authentication contexts to authentication methods. It allows the administrator to control the complete behavior of the MCB:

MCB Configuration File

```
<?xml version="1.0" encoding="UTF-8"?>
<MultiContextBroker>

    <!-- Location of the velocity properties file used by the MCB. Controls where login page templates are
stored. -->
    <velocityPropertiesFile>/opt/shibboleth-idp/conf/velocity.properties</velocityPropertiesFile>
    <!--
        authOptions controls how the MCB responds to an user without an existing session.
        The attribute presentAll set to true tells MCB to present all authentication
        methods to the end user that satisfy the requested authentication contexts.
        If presentAll is false, then the intialAuthMethod value specifies the method
```

```
        presented to the end user for initial authentication.
    -->

    <!--
        Possible first login options:
            1. Show only the context options requested by the SP. If this IdP does not support any of the
               requested methods, then a SAML error will be returned to the SP. If there is only a single
               match between the requested and available authentication contexts, then that context will be
used.
            2. Show a default context list for initial authentication
        Only 1 choice is allowed by the schema (and logically)
    -->
    <initialAuthContext>
        <defaultContexts>
            <context name="urn:oasis:names:tc:SAML:2.0:ac:classes:Password" />
            <context name="http://id.incommon.org/assurance/bronze" />
        </defaultContexts>
        <!--
        <requestedOnly/>
        -->
    </initialAuthContext>

    <!-- The reference to the ID in Shibboleth's attribute-resolver.xml that is used to obtain the list of
valid context levels for a user. -->
    <idms attributeResolverID="assurance" />

    <!--
        The maximum number of failures allowed a user before returning a SAML failure to the
        relying party. Must be specified according to schema definition. Set to a value of -1
        to allow an unlimited number of login failures.
     -->
    <maxFailures>3</maxFailures>

    <!--
        authContexts is the list of configured contexts the MCB will honor.
     -->
    <authnContexts>
        <!--
            For each context, the name attribute is used to match up with the values returned by the IdMS and
also
            used to match the requested authentication context sent by the SP.
            The method attribute corresponds to the authentication method this context uses.
         -->
        <context name="http://id.incommon.org/assurance/bronze" method="password">
            <!--
                Note that since the bronze level allows silver and silver allows gold, means gold is acceptable
here. Contexts
                are inherited. Since two levels of silver have been configured, either is acceptable for
authenticating at the
                bronze level (but only because both are listed).
            -->
            <allowedContexts>
                <context>http://id.incommon.org/assurance/silver</context>
                <context>http://id.incommon.org/assurance/silver-token</context>
            </allowedContexts>
        </context>

        <context name="http://id.incommon.org/assurance/silver" method="strongpassword">
            <!--
                allowedContexts is a list of contexts which satisfy this level as well
             -->
            <allowedContexts>
                <context>http://id.incommon.org/assurance/silver-token</context>
            </allowedContexts>
        </context>


        <context name="http://id.incommon.org/assurance/silver-token" method="token">
            <!--
                allowedContexts is a list of contexts which satisfy this level as well
             -->
```

```
            <allowedContexts>
                <context>edu:internet2:middleware:assurance:mcb:tokenpluspin</context>
            </allowedContexts>
        </context>
        <context name="edu:internet2:middleware:assurance:mcb:tokenpluspin" method="tokenpluspin" />


    </authnContexts>

    <!--
        authMethods is the list of authentication methods supported by the MCB
     -->
    <authMethods>
        <!--
            A method defines one authentication method. The name attribute corresponds to the method value
            used in the context definition. The bean attribute is the name of the bean loaded by the Spring
            framework during Shibboleth startup. The value of the method node is the friendly name
            used for display purposes.

            Each authentication method servlet would be the same as an existing Shibboleth servlet with the
exception
            that control would be returned to the MCB upon completion of each authentication attempt (meaning
the
            servlet could not ask the user for credentials again as a result of a failure, control would go the
MCB first
            and then be sent back to the servlet if still allowed).
         -->
        <method name="token" bean="mcb.token">
            Silver Assurance Level (via hardware token)
        </method>
        <method name="strongpassword" bean="mcb.username2password2">
            Silver Assurance Level
        </method>
        <method name="password" bean="mcb.usernamepassword">
            Username/Password Only
        </method>
        <method name="tokenpluspin" bean="mcb.tokenpluspin">
            Gold Level - Token/PIN Required
        </method>

    </authMethods>
</MultiContextBroker>
```