University of Chicago

The University of Chicago supports both username/password authentication and Multi-Factor using DUO. DUO support for the Multi-Context-Broker is available on GitHub. Note, you need to have Multi-Context Broker version 1.1.2 or better to use the DUO plug-in. All users go through the initial auth using username/password. After that, triggering of DUO depends on whether the SP requested DUO or not. We also support a context where an SP could request that the user either have DUO or be InCommon Silver certified. Yes, this could be handled by having the SP send in both contexts in a preferred order, but we wanted to provide this local context as a convenience.

The idea behind this is that a user who authenticates and has Silver is allowed to SSO through everything except DUO (provided the IDMS allows it). A user who hits an SP which requests DUO will then basically give the user an SSO session through every other context.

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

    <velocityPropertiesFile>/opt/shibboleth-idp/conf/velocity.properties</velocityPropertiesFile>

    <!--
        Show this list of choices for initial authentication to establish a session. Optionally limit the
choices
        to those also requested by the SP. If no choices match the SP request, then show the entire list just
as if
        the SP had not requested any.
    -->
    <initialAuthContext >
            <context name="urn:oasis:names:tc:SAML:2.0:ac:classes:Password" />
    </initialAuthContext>
    <!--
        This value identifies the ID of the attribute in the Shibboleth attribute-resolver.xml file that
contains the user's allowed context values.
    -->
    <idms attributeResolverID="eduPersonAssurance" />

    <!--
        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="urn:oasis:names:tc:SAML:2.0:ac:classes:Password" method="password">
            <allowedContexts>
            </allowedContexts>
        </context>

        <context name="http://uchicago.edu/duoorsilver" method="duo">
                <allowedContexts>
                        <context name="http://id.incommon.org/assurance/silver" />
                </allowedContexts>
        </context>
        <context name="http://uchicago.edu/duo" method="duo">
                <allowedContexts>
                </allowedContexts>
        </context>

        <context name="urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport" method="password">
                <allowedContexts>
            </allowedContexts>

        </context>
```

```
        <context name="http://id.incommon.org/assurance/silver" method="silver">
            <!--
                allowedContexts is a list of contexts which satisfy this level as well
             -->
            <allowedContexts>
                <context name="urn:oasis:names:tc:SAML:2.0:ac:classes:Password" />
                <context name="http://uchicago.edu/duo" />
            </allowedContexts>
        </context>

    </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 submodule bean loaded by
            the Spring framework during Shibboleth startup. The value of the method node is the friendly name
            used for display purposes.
        -->
        <method name="password" bean="mcb.usernamepassword">
            Username/Password Only
        </method>

        <method name="silver" bean="mcb.usernamepasswordsilver">
            Silver Assurance Level
        </method>
        <method name="duo" bean="mcb.duo">
            Duo
        </method>
    </authMethods>
</MultiContextBroker>
```