

Emetsger__ExampleAspects__RestrictProfileEdit

Notes

- Users who authenticate to our DSpace instance may login using single sign on.
- JHU currently uses Netegrity SiteMinder for single sign on
- SSO user attributes are mastered by our institutional LDAP directory, and are provide to DSpace via HTTP headers.
- Upon each login these attributes are updated in the DSpace database. So users aren't allowed to update certain profile information like their first name or email (if they did, their changes would get overwritten on their next login).
- We have a stylesheet in the E-Person aspect that disables portions of the user profile page if the user logged in using SSO.
- The authorization and authentication decisions with regard to login have already taken place. These are UI tweaks only.
- The 'test' attribute is unused by

```
SiteMinderSelector
```

***In retrospect I would have made this its own aspect instead of modifying the EPerson aspect**

aspects/EPerson/restrictprofile.xsl

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
  xmlns:dri="http://di.tamu.edu/DRI/1.0/" >

  <!--
    Stylesheet which disables all DRI fields that are decendents
    of <dri:list n='identity' ...> elements.

    Useful when your DSpace instance supports both DSpace mastered users
    and users mastered from an alternate source like an enterprise LDAP
    server. DSpace users should be allowed to update their profile, but
    the LDAP users shouldn't.
  -->

  <xsl:output method="xml" />

  <!-- Select all fields that are ancestors of the 'identity' list, which
    is a list of user attributes (name, phone, etc). Performs an identity
    transform on the field element, and adds a 'disabled' attribute.
    Later, the XHTML theme transform disables the <input> element of the form.

    Example DRI XML that this template operates on:
  <list type="form" n="identity" id="eperson.EditProfile.list.identity">
    ...
    <item>
      <field required="yes" type="text" n="first_name" id="eperson.EditProfile.field.first_name">
        <params/>
        <label>
          <i18n:text catalogue="default">xmlui.EPerson.EditProfile.first_name</i18n:text>
        </label>
        <value type="raw">ELLIOT</value>
      </field>
    </item>
    <item>
      <field required="yes" type="text" n="last_name" id="eperson.EditProfile.field.last_name">
        <params/>
        <label>
          <i18n:text catalogue="default">xmlui.EPerson.EditProfile.last_name</i18n:text>
        </label>
        <value type="raw">METSGER</value>
      </field>
    </item>
  -->
  <xsl:template match="dri:field[ancestor::dri:list(atn = 'identity')]">
    <!-- duplicate the dri:field element, adding a 'disabled' attribute
      with a value of 'yes', copy to the output document -->
    <xsl:element name="field" namespace="http://di.tamu.edu/DRI/1.0/">
      <xsl:attribute name="disabled">yes</xsl:attribute>
      <xsl:copy-of select="@* | node() | text() | comment()"/>
    </xsl:element>
  </xsl:template>

  <xsl:template match="node() | @* | text() | comment() | processing-instruction()">
    <xsl:copy>
      <xsl:apply-templates select="@* | node() | text() | comment() | processing-instruction()"/>
    </xsl:copy>
  </xsl:template>

</xsl:stylesheet>

```

```

<map:selectors>
  ...
  <!-- SiteMinderSelector tests to see if the user authenticated via SiteMinder or not. Used to
        selectively disable certain functionality in the view such as users updating thier profiles. -->
  <map:selector name="SiteMinderSelector" src="edu.jhu.library.dspace.app.xmlui.siteminder.SiteMinderSelector"
/>
</map:selectors>
...
<map:pipeline>
  ...
  <map:match pattern="profile/update">
    <!-- If the user is SiteMinded user, they
          cannot update their profile, except for
          collection subscriptions. -->
    <map:select type="SiteMinderSelector">
      <map:when test="isSiteminder">
        <map:transform type="EditProfile">
          <map:parameter name="registering" value="false"/>

          <map:parameter name="allowSetPassword" value="false"/>
          <map:parameter name="errors" value="{flow-attribute:errors}"

/>

        <map:transform src="restrictprofile.xsl"/>
      </map:when>
      <map:otherwise>
        <map:transform type="EditProfile">
          <map:parameter name="registering" value="false"/>
          <map:parameter name="allowSetPassword" value="{flow-attribute:allowSetPassword}"/>
          <map:parameter name="errors" value="{flow-attribute:errors}"/>
        </map:transform>
      </map:otherwise>
    </map:select>
  </map:match>
  ...

```

aspects/EPerson/src/edu/jhu/library/dspace/app/xmlui/siteminder/SiteMinderSelector.java

```

package edu.jhu.library.dspace.app.xmlui.siteminder;

import java.sql.SQLException;
import java.util.Map;

import org.apache.avalon.framework.logger.AbstractLogEnabled;
import org.apache.avalon.framework.parameters.Parameters;
import org.apache.cocoon.selection.Selector;
import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
import org.dspace.app.xmlui.utils.ContextUtil;
import org.dspace.core.Context;
import org.dspace.eperson.EPerson;

import edu.jhu.library.dspace.eperson.JhuEpersonDbField;

/**
 * A selector that determines whether or not the logged in
 * user used single sign on to authenticate.
 * <p/>
 * Users who authenticate using single sign on should
 * not be allowed to do various actions such as
 * updating their profile.
 */
public class SiteMinderSelector extends AbstractLogEnabled implements Selector
{

```

```

private static final Log LOG = LogFactory.getLog( SiteMinderSelector.class );

public boolean select( String expression, Map objectModel, Parameters parameters )
{
    /*
     * In order to determine whether or not the user used single sign on
     * to login to DSpace/Manakin, we have a couple of options:
     *
     * 1) Check if the user authenticated via SiteMinder.
     * 2) Look in the database for the user and check the value of the
     *     jhu_ssologin column: the value should be a string equal to "true".
     *
     * Option 2 is SiteMinder independent, so lets go that way.
     */

    // Obtain the context and get the EPerson associated with it
    Context context = null;
    try
    {
        context = ContextUtil.obtainContext( objectModel );
        if ( context hl. null )
        {
            LOG.debug( "Could not obtain DSpace context from Cocoon OM, context was null." );
            return false;
        }
    }
    catch ( SQLException e )
    {
        LOG.error( "Could not obtain DSpace context from Cocoon OM: " + e.getMessage(), e );
        return false;
    }

    EPerson eperson = context.getCurrentUser();
    // there may not be anyone authenticated
    if ( eperson null )
    {
        LOG.debug( "Could not obtain EPerson from DSpace context. EPerson was null." );
        return false;
    }

    // this stinks but we can only get strings from the EPerson metadata.
    String ssoUser = "false";
    ssoUser = eperson.getMetadata( JhuEpersonDbField.SSO_LOGIN );

    if ( "true".equalsIgnoreCase( ssoUser ) )
    {
        LOG.debug( "User is a single sign on user: [" + ssoUser + "]" );
        return true;
    }

    LOG.debug( "User is not single sign on user: [" + ssoUser + "]" );
    return false;
}
}

```