Using the XACML Editor

XACML versus Drupal Permissions

It is important to note that XACML policies act in addition to Drupal policies, rather than replacing them. If a user is granted Object Viewing permissions on a collection that they do not have the proper Drupal permissions (in admin/user/permissions) to access will **not** be able to view that object. Likewise, a user with the "view fedora collection" permission, but no Object Viewing permission on a collection that has XACML Object Viewing Restrictions enabled, will **not** be able to view that collection.

XACML provides access restrictions at a more fine-grained level than Drupal (for instance, blocking access to a single collection, object, or datastream instead of an entire content type), but proper policies must take existing Drupal permissions into account.

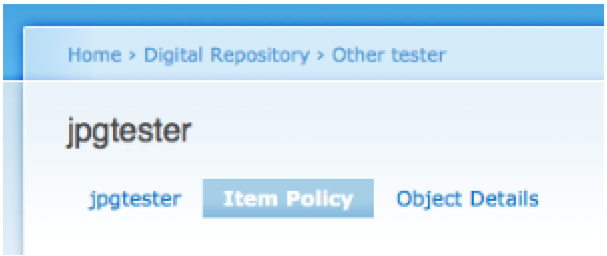
Configuration

A limited set of basic configuration are available by going to Administer and clicking **Islandora XACML** (admin/settings/islandora_xacml). You will find two tabs:

- **Islandora XACML API -** Define Solr RELS-EXT (default settings should work)
- **Islandora XACML Editor -** Define options available in the XACML editor for all collections.

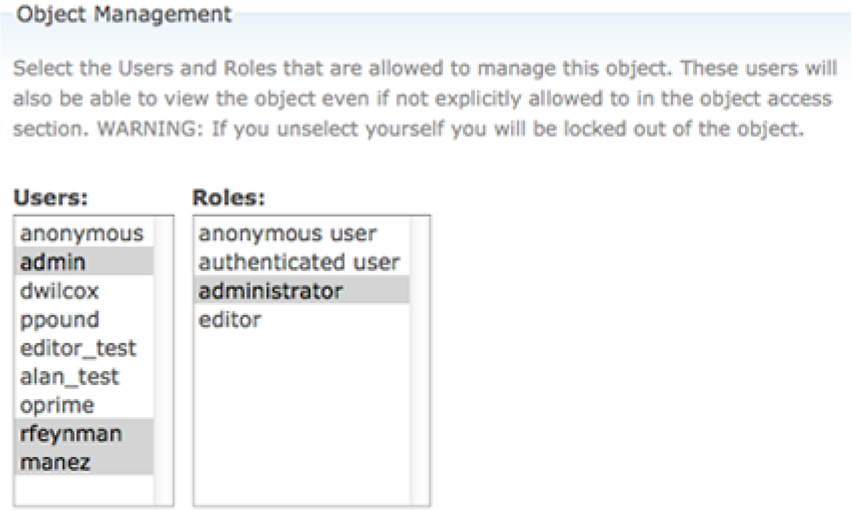
Item/Child Policy

With the XACML Editor enabled, each object and collection will gain a new tab where you can define XACML policies for that object/collection. At the object level this tab is Item Policy; at the collection level, it is Child Policy (defining policies for all children of that collection). The basic options under both tabs are similar, with additional configuration options available for Collections.



Object Management

Object Management policies effect who can set XACML policies for a particular object. Anyone who can Manage an object can also view it, even if Object Viewing permissions would otherwise deny access. To select multiple users, use ctrl+click (Windows) or command+click (Mac).



Object Management

Select the Users and Roles that are allowed to manage this object. These users will also be able to view the object even if not explicitly allowed to in the object access section. WARNING: If you unselect yourself you will be locked out of the object.

Users:

anonymous
admin
dwilcox
ppound
editor_test
alan_test
oprime
rfeynman
manez

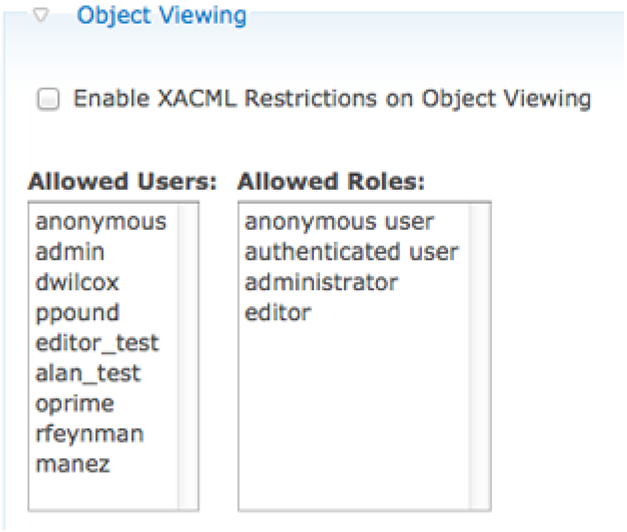
Roles:

anonymous user
authenticated user
administrator
editor

In order to prevent accidentally locking yourself out of an object or collection, the XACML Editor will prompt you to always select your account and that of the admin user (user 1). To remove a XACML policy completely, delete the *Xacml Policy Stream* under the Object Details tab rather than deselecting members in the XACML Editor.

Object Viewing

Object Viewing policies control who can view an object. If this option is not enabled, then only regular Drupal permissions will apply. When enabled, this option will override Drupal permissions negatively, but not positively; in other words, a user who has Drupal permissions to view an object but not XACML permissions will **not** be able to view that object, and a user who does not have Drupal permissions but does have XACML permissions will also **not** be able to view the object. In order to view the object, the user will need **both** Drupal and XACML permissions to access it.

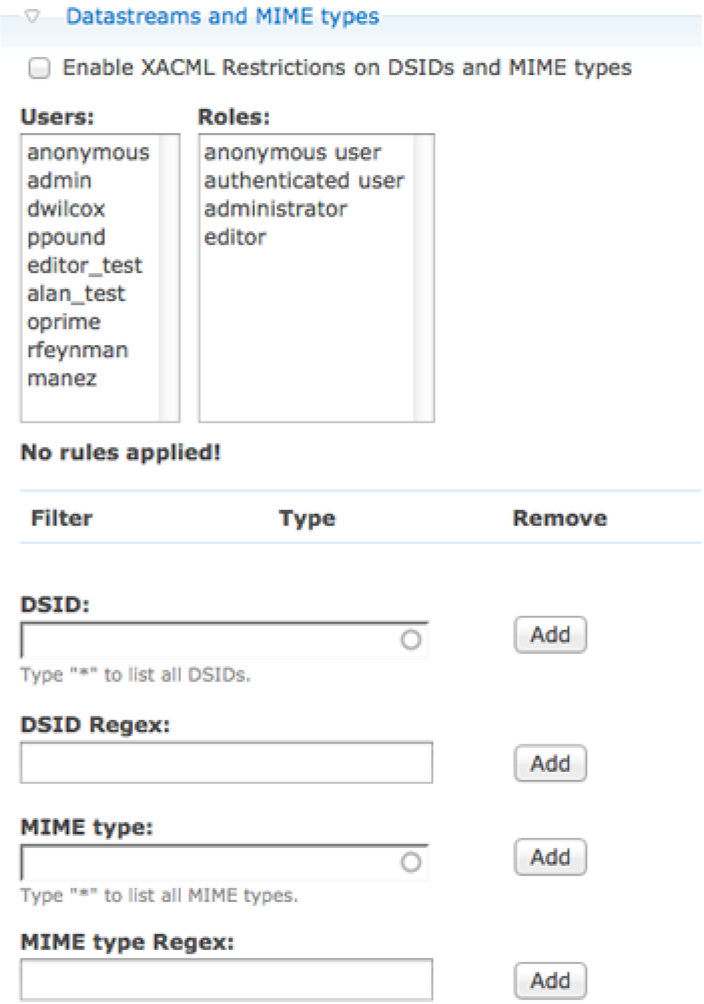


Datastreams and MIME types

Datastream (DSIDS) and MIME type restrictions control user access to individual data streams on an object or collection. This restriction applies to viewing those datastreams, and not to modifying them. Permissions to modify datastreams should be controlled l through Drupal permissions in admin/user/permissions. If this option is enabled, users who do not have permission to view certain datastreams will not see them listed for an object or collection.

Restrictions in this section must be enabled by DSID or MIME type, instead of simply being applied to the entire object.

- **DSID:** Restrict a particular data stream on the object. Provided as a lookup field so that you can search for available data streams.
- **DSID Regex:** Create a rule to restrict all data streams fitting a certain pattern or in a certain class, i.e, POLICY/*
- **MIME type:** Restrict access to a particular MIME type on an object. Provided as a lookup field so that you can search for the MIME types available.
- **MIME type Regex:** Create a rule to restrict all MIME types fitting a certain pattern or in a certain class, i.e, text/*



▽ Datastreams and MIME types

□ Enable XACML Restrictions on DSIDs and MIME types

Users:	Roles:
anonymous	anonymous user
admin	authenticated user
dwilcox	administrator
ppound	editor
editor_test	
alan_test	
oprime	
rfeynman	
manez	

No rules applied!

Filter	Type	Remove

DSID:

Type "*" to list all DSIDs.

DSID Regex:

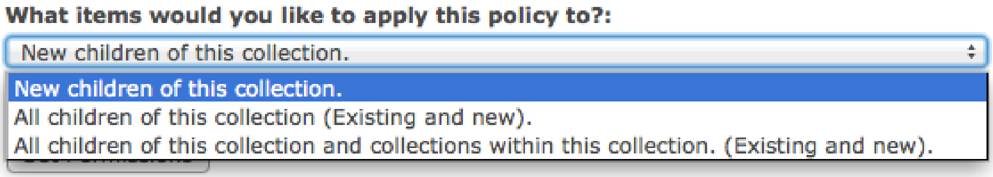
MIME type:

Type "*" to list all MIME types.

MIME type Regex:

Collection Children

When editing policy at the collection level, an additional option is available to determine how the policies will be applied to children of the collection (objects and child collections). If there are numerous objects in the collection or its child collections, this process may take some time.



What items would you like to apply this policy to?: New children of this collection.

New children of this collection.
All children of this collection (Existing and new).
All children of this collection and collections within this collection. (Existing and new).