Islandora Checksum Checker

Overview

This module verifies the checksums derived from Islandora object datastreams and adds a PREMIS 'fixity check' entry to the object's audit log for each datastream checked. Please note that adding this entry updates the object (specifically, it changes the object's lastModifiedDate).

Islandora Checksum Checker needs to be periodically triggered using Drupal's cron functionality (or an alternative like Elysia Cron, or using an operating-system-level scheduler like Linux's crontab to run a drush command (documented below).

With each run, the module performs checksum verification on a configurable list of object datastreams. When it has checked the datastreams of all objects (from oldest to newest), it will start from the beginning (i.e. with the oldest object in your repository) and repeat the verification cycle.

Dependencies

This module requires the following modules/libraries:

- Islandora
- Tuque
- Islandora Checksum (Optional)

This module is only useful if you use Fedora Commons to generate checksums on datastreams. The easiest way to have Fedora Commons generate checksums is to install and enable the Islandora Checksum module.

Downloads

Release Notes and Downloads

Installation

Install as usual, see this for further information.

Configuration

Set the cron method, number of objects to check per cron run, datastream to check, who to sent report to in Administration » Islandora » Checksum checker (admin/islandora/checksum_checker).

blocked URL

The two most common options for scheduling the verification are:

1. choose 'Drupal cron' in the module's admin settings and make sure that you have cron running on your site, and
2. choose 'drush script' in the module's admin settings and set up a scheduled job using a utility like Linux's cron to trigger the verification via drush.

Option 1 is simplest because it requires no additional configuration but only works if all of the objects in your Islandora repository are viewable by the Drupal 'anonymous' user (since Drupal 7's cron runs as anonymous).

Option 2 is necessary if any of your objects are not viewable by anonymous. It also has the advantage of running the verification process independently from other tasks initiated be Drupal's built-in cron.

The drush command you need to run is `drush run-islandora-checksum-queue`. You should include drush's --root and --user options to define the path to your Drupal installation's root, and an Islandora user account that has privileges to view all datastreams, respectively. A typical Linux crontab entry (in this case, to run every hour) is:

```
0 * * * * /usr/bin/drush --root=/path/to/drupal --user=fedoraAdmin run-islandora-checksum-queue
```

Frequency of verification

How often you should run this command will depend on several factors, including how many objects you have in your Islandora repository and how many days or months you will tolerate between reverification of the same object's datastream checksums.

Assuming that you configure this module to check 50 objects every time it runs and that you have 10,000 objects in your Islandora repository, all objects will be checked every 8 days if you configure it to run every hour. If you configure this module to run every 6 hours, all objects will be checked every 50 days.

Also, since the results of the verification are recorded in each object's audit log, the more often you verify checksums, the larger the audit logs (and therefore the objects themselves) become. Each time a datastream is checked, the object's audit log grows by about 450 bytes. An object that has five datastreams that are all being checked will grow by about 4.5 kB/month if it is checked twice a month. A 10,000-object repository will use about 43 MB of disk every month just to store the results of routine checksum verification in the objects' audit logs.

In addition, each time a datastream's checksum is verified, about twice as much data is written to your fedora.log as is stored in the object's audit log, so a more realistic estimate of how much disk space is consumed by routine checksum verification is three times the figures calculated above.