

Clowder Test Plan

- **Release 1.5**
 - **Bug Fixes (Highest Testing Priority)**
 - **Permissions**
 - CATS-881: In a private Clowder instance, a superadmin can now see datasets in a space that he/she is not part of
 - CATS-887: if permissions are not private we should not list all datasets
 - **Fixes to Authentication Providers**
 - CATS-868: If no local password and only 1 provider, redirect to the provider login page immediately
 - CATS-911: fixed ldap userid
 - **Fixed MongoDB Dataset Service findFileById**
 - CATS-897 / CATS-906: mongodbdatasetsevice.findbyfileid will also return datasets indirectly containing give fileid
 - **Fixed RabbitMQ Parameter Escaping**
 - CATS-905: Parameters subdocument is escaped in rabbitmq message
 - **Fixed Extractors Logging Sensitive Data in Plaintext**
 - CATS-903: Fixed extractors printing the private key in error messages
 - **New Views: ExtractorInfo / ExtractorDetails**
 - CATS-891: New global extractor view (e.g. table)
 - CATS-892: New Extractor details view
 - **New API: Reverse Proxy**
 - CATS-793 / CATS-889: Clowder Reverse Proxy API - GET/POST/PUT/DELETE
 - CATS-895: Clowder Reverse Proxy API - Authentication
 - CATS-910: Fixes to swagger spec
 - Notes
 - **New Features**
 - **Tracking Usage Metrics**
 - CATS-374 / CATS-375: Track usage metrics - files/datasets + views/downloads
 - **Per-Space Extractors**
 - CATS-890: Per-space extractor list should now show more detail (e.g. table)
 - CATS-799: Extractors should now respect per-space enabled/disabled options
 - **Configuration / Maintainability Changes**
 - CATS-884: Added Orcid Id's to zenodo.json file
 - BD-2128: Add overridable environment vars for Mongo and Rabbit MQ URIs
 - Script to create service accounts

Release 1.5

Bug Fixes (Highest Testing Priority)

Permissions

- **CATS-881: In a private Clowder instance, a superadmin can now see datasets in a space that he/she is not part of**
 1. **SETUP:** Is there special setup or configuration needed to make an instance "private"?
 2. Login to Clowder
 3. Make yourself SuperAdmin using the dropdown at the top-right
 - a. You should now be able to see all datasets on Clowder, including those that you are not a part of
- **CATS-887: if permissions are not private we should not list all datasets**
 1. **SETUP:** Is there special setup or configuration needed to make an instance "private"?
 2. Login to Clowder
 3. ???
 - a. Expectations?

Fixes to Authentication Providers

- **CATS-868: If no local password and only 1 provider, redirect to the provider login page immediately**
 1. ???
- **CATS-911: fixed ldap userid**
 1. Enable LDAP login in Clowder
 2. Login with one user via LDAP
 3. Login with a second user via LDAP
 4. ???
 - a. Expectations?

Fixed MongoDB Dataset Service findFileById

- **CATS-897 / CATS-906:** `mongodbdatasetsevice.findbyfileid` will also return datasets indirectly containing give fileid
1. add file under path `test/a/b/file`.
 - a. run get paths endpoint on this file.
 - b. delete this file from clowder.
 - c. **Expectations?**
 2. add file under test dataset.
 - a. run get paths endpoint on this file.
 - b. delete this file from clowder.
 - c. **Expectations?**

Fixed RabbitMQ Parameter Escaping

- **CATS-905:** Parameters subdocument is escaped in rabbitmq message
1. Enable the RabbitMQ plugin in your Clowder instance
 2. Register an extractor with Clowder
 3. Manually submit a file or dataset for extraction
 4. View the message in RabbitMQ 
 - a. You should see that the map of "parameters" is no longer escaped as a string

Fixed Extractors Logging Sensitive Data in Plaintext

- **CATS-903:** Fixed extractors printing the private key in error messages
1. Enable the RabbitMQ plugin in your Clowder instance
 2. Register an extractor with Clowder
 3. Manually submit a file or dataset for extraction
 - a. *Hint:* Try to find some case that you know will produce an extractor error
 4. Check the extractor logs / metadata
 - a. You should no longer see the `secret_key` explicitly printed to the logs during errors

New Views: ExtractorInfo / ExtractorDetails

- **CATS-891:** New global extractor view (e.g. table)
1. Enable the RabbitMQ plugin in your Clowder instance
 2. Start an extractor that has the process block defined in `extractor_info.json`
 3. Login as an admin user via <http://localhost:9000/login>
 4. Expand the "Admin" dropdown menu at the top-right (gear icon)
 5. Choose the new option for "Extractors" (not to be confused with "Extractions", which previously existed)
 - This should navigate you to the new <http://localhost:9000/extractors> view
 - By default, no extractors should be enabled
 - **CATS-908:** You shouldn't see any extraneous `<br>` elements between the page header and the table
 6. Enable an extractor and click "Update"
 - You should be brought back to the Clowder home page
 7. Navigate back to the "Extractors" view and refresh the page
 - You should see that your selection is still enabled
 - The refresh further confirms that this change has been persisted to MongoDB (not just in-memory)
 8. Navigate to a per-space extractor list for a space (e.g. `spaces/:spaceId/extractors`)
 - **CATS-909:** You should see that the extractor that was enabled globally cannot be unchecked
- **CATS-892:** New Extractor details view
1. Enable the RabbitMQ plugin in your Clowder instance
 2. Start and register an extractor with Clowder (for example, `nlsa.image.metadata`):

```
export PUBLIC_IP=<YOUR_PUBLIC_IP>
export CLOWDER_API_KEY=<CLOWDER_API_KEY>
docker run -it --rm --net=host -e RABBITMQ_URI="amqp://guest:guest@localhost/%2f" -e
REGISTRATION_ENDPOINTS="http://${PUBLIC_IP}:9000/api/extractors?key=${CLOWDER_API_KEY}" clowder
/extractors-image-metadata
```

3. Navigate to `localhost:9000/extractors/extractorName` (for example: <http://localhost:9000/extractors/nlsa.image.metadata>)
 - a. You should see a new view that roughly matches the mockups [shown here](#)
 - b. The new view should list all/most of the common fields of the underlying `ExtractorInfo` object (note that these fields were previously hidden everywhere else in the UI)

New API: Reverse Proxy

- **CATS-793 / CATS-889: Clowder Reverse Proxy API - GET/POST/PUT/DELETE**

1. Configure the following in your custom.conf, then run Clowder using that configuration:

```
clowder.proxy {  
  geopub="https://geoserver.ncsa.illinois.edu/geoserver"  
}
```

2. Without logging in, attempt to hit the geoserver API through the proxy - for example: http://localhost:9000/api/proxy/geopub/clowder/wms?service=WMS&version=1.1.0&request=GetMap&layers=clowder:tmppFKK2_.zip12172744&styles=&bbox=365741.5,4434859.5,375398.5,4444630.5&width=506&height=512&srs=EPSG:26916&format=image/png
 - You should get a generic response of *Not authorized*
3. Log into Clowder via <http://localhost:9000/login>
4. Attempt to hit the geoserver API again using the proxy - for example: http://localhost:9000/api/proxy/geopub/clowder/wms?service=WMS&version=1.1.0&request=GetMap&layers=clowder:tmppFKK2_.zip12172744&styles=&bbox=365741.5,4434859.5,375398.5,4444630.5&width=506&height=512&srs=EPSG:26916&format=image/png
 - This should return the same response/image as the original endpoint: https://geoserver.ncsa.illinois.edu/geoserver/clowder/wms?service=WMS&version=1.1.0&request=GetMap&layers=clowder:tmppFKK2_.zip12172744&styles=&bbox=365741.5,4434859.5,375398.5,4444630.5&width=506&height=512&srs=EPSG:26916&format=image/png
5. Choose an endpoint_key that does not exist - for example: <http://localhost:9000/api/proxy/ThisIsNonsense>
 - You should get a generic response of *Not found: ThisIsNonsense*

- **CATS-895: Clowder Reverse Proxy API - Authentication**

1. Configure your clowder.proxy as follows:

```
clowder.proxy {  
  authtest="http://username:12345@httpbin.org/basic-auth/username/12345"  
}
```

2. Navigate to <http://localhost:9000/api/proxy/authtest>
 - You should get a generic message from Clowder saying *Not authorized*
3. Navigate to <http://localhost:9000/login> and login to Clowder
4. Navigate once more to <http://localhost:9000/api/proxy/authtest>
 - This time, you should be authenticated into the test service using the service account credentials specified in the proxy configuration

- **CATS-910: Fixes to swagger spec**

1. view the updated spec in Swagger UI here: <https://clowder.ncsa.illinois.edu/swagger/?url=http://localhost:9000/swagger>

Notes

You can also serve swagger.yml some other way, if you'd like, but CORS did (I think) prevent me from linking Swagger UI directly to a raw file in our OpenSource BitBucket.

For example, this did not work: <https://clowder.ncsa.illinois.edu/swagger/?url=https%3A%2F%2Fopensource.ncsa.illinois.edu%2Fbitbucket%2Fprojects%2FCATS%2Frepos%2Fclowder%2Fraw%2Fpublic%2Fswagger.yml%3Fat%3Drefs%252Fheads%252Fbugfix%252FCATS-910-fix-swagger-documentation-format>

New Features

Tracking Usage Metrics

- **CATS-374 / CATS-375: Track usage metrics - files/datasets + views/downloads**

1. Enable the RabbitMQ plugin in your Clowder instance
2. Register the standard image preview extractor with Clowder
 - a. **NOTE:** This will prevent "views" of a file from being counted as a download, which could be confusing to users
3. Login to Clowder
4. Upload a new file
5. View it in Clowder
 - a. You should see that the file has 1 view and 0 downloads
6. Refresh the page a few times
 - a. You should see that the "views" count is incremented for each refresh
7. Download the file and refresh the page again
 - a. You should see now that the file has 1 download

Per-Space Extractors

- **CATS-890: Per-space extractor list should now show more detail (e.g. table)**

1. Enable the RabbitMQ plugin in your Clowder instance
2. Start an extractor that has the process block defined in `extractor_info.json`
 - a. For example: `image-metadata`
3. Create a space (or navigate to an existing space)
4. Admins should see an "Extractors" button/link on the right of the `spaces/:spaceId` view - click this link
 - a. You should be brought to `spaces/:spaceId/extractors`
 - b. You should see any registered extractors listed here
 - c. This view now should roughly match the mockups attached to this ticket (sans the "Show Columns" dropdown)
 - d. Each extractor should list the name, description, author, and process triggers, as well as the checkbox to enable or disable them
 - e. **CATS-908:** You shouldn't see any extraneous `<br>` elements between the page header and the table

- **CATS-799: Extractors should now respect per-space enabled/disabled options**

1. Start from the end of the previous test case (CATS-890)
 - a. Make sure that the `image-metadata` extractor is **disabled**, both globally and in your test space
2. Upload an image file to your test space
 - a. Your extractor should **not** trigger, and should remain idle
3. Enable the extractor globally via the new global extractor view (e.g. `/extractors`)
4. Upload another test image
 - a. This time your extractor should fire, since it has been enabled globally
5. Disable the extractor globally (via `/extractors`)
6. Enable the extractor at the space-level (e.g. `spaces/:spaceId/extractors`)
7. Upload one more test image
 - a. The extractor should trigger again, since it has been enabled in this space

Configuration / Maintainability Changes

- **CATS-884: Added Orcid Id's to `zenodo.json` file**

1. ???

- **BD-2128: Add overridable environment vars for Mongo and Rabbit MQ URIs**

1. ???

- **Script to create service accounts**

1. ???