Rucio - S3-compatible access interface for analysis facilities

- Fellow: Kyrylo Meliushko
- Project Mentors: Martin Barisits, Matthew Feickert, Lukas Heinrich, Mario Lassnig

Project Description

Rucio is an open source software framework that provides functionality to scientific collaborations to organize, manage, monitor, and access their distributed data and dataflows across heterogeneous infrastructures. Rucio was originally developed to meet the requirements of the high energy physics experiment ATLAS, and is continuously enhanced to support diverse scientific communities. Since 2016 Rucio has orchestrated multiple exabytes of data access and data transfers globally.

Considering the growth of Rucio use cases, there is a continuous need to implement new data exchange features. One example is integrating the Amazon S3 protocol with Rucio's object-storage architecture, which was done during LHC Run-2. This implementation already provides the user with the essential features, but there is also a need for a more flexible and streamlined data exchange process that allows scientists to efficiently share specific data subsets for collaboration and analysis purposes. A key aspect of this project is therefore to introduce an interface that enables users to directly manage S3 buckets. This enhancement would greatly improve the user experience, providing a seamless way to handle and manipulate specific pieces of data produced within Rucio for efficient collaboration and analysis. Allowing users to store data products produced in their research in a portable, shareable and standardized way.

As the connection between Rucio and S3 already exists, though without a public API, exposing direct access from the user interface (UI) to the S3 API should not be problematic. However, a few additional requirements need to be considered. The buckets created using the implemented interface should seamlessly integrate with other available S3 clients, such as MinIO's "mc". Each user should have the ability to manage their own buckets, as well as access shared buckets related to the experiment for convenient data exchange. The functionality to create own scopes and access other users' scopes should also be implemented. Since some users are already authenticated to use Rucio, exporting the credentials for expanded usage via "mc" or other tools should be feasible. Additionally, it is important to provide unit tests to ensure successful data transfer and verify proper access control.

Deliverables

The desired software deliverables include the capability to create user-level and group-level buckets for organizing data storage. This allows users to efficiently manage their data by creating distinct buckets at different levels. To allow such functionality, S3 credentials have to be provided, which should be implemented by allowing users to export them from Rucio. Interaction with the storage should be possible using both S3 API and Rucio API.

Timeline

- **Weeks 1-3:**
  Familiarize myself with the Rucio and MinIO's "mc" tool documentation and source code. Explore possible ways to handle the authentication, and go through S3 API.. Prepare a development environment by forking the main Rucio repository. Open a GitHub Issue describing the goal of this project and steps to achieve it according to contributing guidelines.
- **Weeks 4-5:**
  Work on the first MVP, test data transfer capabilities without any extra features. Combine it with the existing Rucio command line interface. Write unit-tests to define what should be reached by the end of the project.
- **Weeks 5-7:**
  Test the integration with other available S3 clients, such as MinIO's "mc". Ensure that it is possible to extract valid credentials from the current Rucio connection to use them with "mc". Start a detailed documentation for the developed extension, providing clear instructions and examples that showcase its usage.
- **Weeks 7-10:**
  Work on the additional features on Rucio's side. Ensure each user can create buckets writable by themselves and/or other members of selected experiments, share files with other teams, and have a configurable quota of managed objects. Granting access to bucket's under one's scope to other users should be possible.
- **Weeks 10-12:**
  Close the issue in Rucio's GitHub repository after making sure the extension works correctly. Refine the documentation and ensure its clarity, prepare a final presentation with all completed work and detailed motivation behind it.

References

- https://hepsoftwarefoundation.org/gsoc/2022/proposal_Rucio-S3.html
- https://rucio.cern.ch/
- https://rucio.cern.ch/documentation/contact_us/