EOS workshop

Contribution ID: **34** Type: **not specified**

CloudStor Minio: Improving S3 performance in CloudStor

Monday 4 February 2019 16:00 (20 minutes)

We at AARNet as well as the research community in Australia need bulk data access to our synch servers because one-off ingest of seriously large datasets performs subpar across the webdav/synch pathway. This presentation will discuss AARNet's experiences, journey and many iterations to achieve high-speed data transfers via S3 protocol (the de facto standard) and the challenges and improvements made along the way.

Minio helps some users interact with CloudStor using the S3 protocol. In the beginning we mounted EOS, CloudStor's storage backend via FUSE and ran Minio over the top. We found that by using this approach transfers were very slow for large files and our FUSE mount kept crashing due to overloaded metadata queries. Fortunately Minio is Open Source and written in the Go programming language which means we can start hacking it!

We then modified Minio to locally stage file uploads, then use xrootd's xrdcopy command as a background task which (from the users point of view), increased uploads from ˜4mb/s to ˜800mb/s. Later on we had a user group uploading a dataset with many tiny files (>100,000) in the one bucket. This uploaded without issue, but doing an object listing on the bucket took over 2 hours. We then modified Minio again so that file listing was done via EOS's /proc/user interface rather than listing via the EOS FUSE mount reducing the time down to 40 seconds. This worked well but the code was no longer maintainable with modifications all over the code base.

From here we made the decision to start over, but rather than hacking the Minio code, we instead decided to write a separate EOS gateway module for Minio. The goal was to fork Minio and improve it to work with EOS in a way that is easy to maintain and update. The other goal of the EOS gateway was to remove the need to use EOS's FUSE connector as it is a source of slowness.

The EOS Gateway for Minio communicates to EOS via EOS's WebDAV, EOS's /proc/user web services and xrootd's, xrdcopy. We also added ownCloud hooks so that files coming in and out of S3 are file scanned so that users can share uploaded data with other users and groups.

AARNet's Minio modifications provides an S3 implementation that allows for tighter integration to Open Source and commercial products that are already integrated in the users'workflow. These includes collaboration and backup products such as FigShare, LabArchive, Alfresco, Commvault to name just a few. This enables users to upload, download, view and share as well as use S3 on the same data area.

Primary author: D'SILVA, Michael (AARNet)

Presenter: D'SILVA, Michael (AARNet)

Session Classification: EOS Ecosystems