North Dakota State University Grouper Contribution

NDSU is migrating from a custom engineered IAM solution to one that utilizes Grouper and midPoint. We're doing this quickly for a migration of such a scale, so our utilization of each might not be the most ideal. However, Grouper is getting the job done well for us. Our legacy system made heavy use of RabbitMQ, so our Grouper deployment is slotting in to use that.

We use a set of Groovy scripts to process SIS and HRMS data from those systems and store that in a MariaDB database. Grouper loaders load information from that DB to populate loader groups, as is pretty standard.

At this point in time, we are not using the provisioning engine, and instead tying Grouper via RabbitMQ into our existing distributed provisioners.

To drive our processes, we have nine+ different group types that are exposed to other systems to use, mostly through RabbitMQ messages. To create these group types we have a custom web application that provides our base template, sends a message through RabbitMQ, which is processed by a Groovy daemon that generates GSH, which is then executed. That same application will show the AD admins which groups are in AD, but don't have an associated Grouper setup, allowing them to pretty much one click to set it up. A lot of our AD groups need to have additional AD configuration on them, so it makes sense to have the Groups from from domain admins into IAM, rather than the other way around.

A lot of our process runs via a Groovy daemon that watches RabbitMQ for messages from Grouper. If a group looks slightly interesting (dropping systemOfRecord, includes, excludes, etc), it will query Grouper to get attribute assignments on the group to see how to proceed. Guava caches those for a period of time to reduce load.

Some of the nine different types are collapsed below to ignore very specific implementation details.

1. AD Groups/Department share - we are slotting this solution in to our existing processes quickly. This may not be the most ideal, but it is working for. Custom template application will query AD and Grouper, and find AD groups in an OU that aren't in Grouper yet. Domain Admins will be able to pretty much do a one click create of the Grouper layout. Groovy message daemon routes these messages as a drop in replacement for legacy system. The AD code can serve out of both IAM systems for different groups during migration.
2. AD Owners - this is just a standard owners group on an AD group, but this will be retransmitted through RabbitMQ so that a description field on the AD group is updated to match owners, making it easier on our domain admins.
3. CMS Authors/Publishers - We run our Typo3 instance with two levels of permissions per workspace. Each is marked and will be retransmitted through the Groovy daemon
4. Entitlements - Flung back out via RabbitMQ to be added to AD user using Powershell
5. LISTSERV - We perform a full population of various LISTSERV lists each morning. In this case Groovy code crawls that app stem looking for groups with that marker, populating the named list in the attribute with members. We have an additional attribute on lists where necessary to populate addresses that aren't Grouper subjects.
6. Other - Catch all of everything else. For instance we populate basis groups with student fee information, and use application groups to setup what becomes permissions to use facilities on campus. For example, staff can take classes, but don't pay fees, thus they can't use the gym for free.

To control the subjects Grouper can see, we have a resource in midPoint that populates the Grouper subject table.

On the other side we're using Grouper groups to populate midPoint orgs. Those orgs are used to create accounts (using a different bridge into the old system). An account in our system is anything that has data associated with it, or it cares about anything beyond a simple immutable user ID. midPoint does a very good job tracking changes to user accounts, and thus that is how it is used.

Additionally, we have a custom application setup that crawls a basis group populated with instructors that are scheduled in rooms. Members of IT can then use that to send targeted messages to people that are teaching in very specific rooms. This application is seeing good use in Fall of 2020 as they need to make changes in each room to accommodate changes due to COVID.