Tues 2pm SantaBarbara

Scribing Template --Tues., Nov 12, 2013 at 2pm -- Santa Barbara

TOPIC: Failsafe Grouper

CONVENER: Jim Fox

SCRIBE: Mike Grady

# of ATTENDEES: 13

MAIN ISSUES DISCUSSED: Failsafe Grouper

- Architecture of typical Grouper deployment
- API for Grouper, more use because of using richer info from Grouper (permissions), ease of access (REST), OAuth styles, etc.
- Should have an Infinite series of replicas of DB, rather than single instance
- What should these replicas look like?
    - "mini" Groupers
    - could be de-normalized, to optimize for "is a member of" query .(Query to Grouper database itself averages 300ms, to their current LDAP thru API front end is 40ms. Haven't measured the performance yet of this proposed approach.)
- need an API layer that has smarts as to which grouper instance to send query to, I.e. a "router" function
- a single de-normalized form that could well accommodate any typical query
- what technologies are being looked at for these horizontal replicas?
    - Elastic Search, based on Lucene, team at U Washington looked at variety of options, like this best so far as far as meeting the needs they foresee for this
    - code that generates elastic search query from REST Grouper query
    - also looking at as cache for student data

Problem with this setup would be some of the more complicated queries? U Washington is basing this on their experience with Groups service use, and has found that by far the queries come down to a few several basic queries.

Is Washington using the new permissions/roles stuff? Not yet. But JIm is confident that if queries of such become common, then the same approach will work for such.

Grouper client does some load balancing today. Some thought that this is too much work beyond what Grouper already does. Others think that this actually seems simpler, easier to understand what is going on and to debug if something is going wrong.

What about LDAP? Couldn't RESTful API just front LDAP servers? (In fact, it is at Washington today.) What's the issue with that approach that you are already doing today?

- Issues with LDAP are it doesn't always do replication well
- Doesn't keep up with changes well (at least with PSP provisioning of LDAP)
- Doesn't handle large groups well (OpenLDAP, but older version/older servers, so maybe investing in new stuff would help.)
- Part of this is management decision where Washington would like to invest in as far as expertise etc.

For a non-DB centric shop (IdM group), this is much more heavily dependent on databases than they would be comfortable with.

How does the security work with this approach?

- Give me all the groups that a user is a member of, do you get all the results, and then filter the results based on the permissions of the caller as to which groups they can see? Results are filtered as they are collected. Same access protection as Grouper, but does it slightly differently.
- Washington does not have "view" controls on the descriptions of the groups, don't worry about that, and no complaints so far

ACTIVITIES GOING FORWARD / NEXT STEPS:

If slides are used in the session, please ask presenters to convert their slides to PDF and email them to acamp-info@incommon.org

Documentation on Univ Washington use of replica (cache) DBs can be found on:

* http://wiki.cac.washington.edu/display/infra/UW+Groups+Service+2.0+Architecture*