COmanage Call 23-July-2010

COmanage Call 23-July-2010

Attending

Heather Flanagan, Independent (chair)
Ken Klingenstein, Internet2
Bob Morgan, U. Washington
Benn Oshrin, Internet2
Steven Carmody, Brown
Chris Hubing, Pennsylvania State U.
Jim Leous, Pennsylvania State U.
Dan Pritts, Internet2
Ann West, Internet2
Renee Frost, Internet2
Steve Olshansky, Internet2
Emily Eisbruch, Internet2, scribe

New Action Items

AI (Benn) will start a discussion on the list about standards and terminology and begin to develop a glossary on the wiki.https://spaces.at.internet2.edu/display/COmanage/Glossary

AI (Heather and SteveO) will work on updating the COmanage website and wiki.

Carry Over Action Items

AI (SteveO) will follow up with Ken about email list structure.

AI (Ken) will develop a COmanage Service Manager job description.

AI (Ken) will email the group a COmanage entrance interview/survey for a VO.

AI (Steven and Benn) will continue to work on demo development.

AI (Ken) will initiate discussion with UK around collaboration platform work.

AI (Heather) will raise the VO Cookbook issue on the upcoming International Collaboration Call.

AI (Jim) will send the group screenshots of the Penn State Confluence dashboard.

AI (Benn, Jim, Chris, Steven) will discuss what's needed to move ahead with the Women's Earth Science Network VO. Status as of 23-July-2010: a conference call is being planned

DISCUSSION

Vision and Strategy Discussion

Benn reviewed his slides presenting an overview of COmanage vision and strategy.

https://spaces.at.internet2.edu/display/COmanage/Presentations

Comments:
• Heather noted the slides will be very helpful as we start to form technical teams with large VOs.
• Ken liked the slides very much. Ken said the term "collaboration management platforms" is better than the term "VOMS" due to historical use of the term VOMS in the grid community. Ken sent these links

http://www.globus.org/grid_software/security/voms.phphttp://edg-wp2.web.cern.ch/edg-wp2/security/voms/voms.html

• Bob noted that when COmanage was first envisioned, the term VO was intentionally not used, so he agrees that we should not use the term VOMS. We should stick with the "CO" term.
• Danno suggested using the term console instead of "Frontend"

Benn stated that he is happy to rename any elements in the slides. He suggested that a dictionary of terminology would be useful.

AI (Benn) will start a discussion on the list about standards and terminology and begin to develop a glossary on the wiki.https://spaces.at.internet2.edu/display/COmanage/Glossary

Benn suggested that the COmanage wiki and website should be updated to reflect the Vision and Strategy slides.

AI (Heather and SteveO) will work on updating the COmanage website and wiki.

Benn noted that concerning the Next Steps outlined in the slides, it will be useful to assign people to the tasks in the near future.

Benn has started moving items into JIRA. He is using the key of "CO" instead of the previous key of "CMG."https://bugs.internet2.edu/jira/browse/CO

Benn believes that the COmanage logo should be fixed so that the gears can move.

Demo

A question was raised about Ken's demo at the GENI meeting in San Diego in July. Ken was no longer on the call, but it was reported that the federated COmanage demo was not complete for the GENI meeting. Ken may have presented COmanage slides; a report on this could be a topic for him on a future call.

Registry of Domesticated Applications

Heather and Benn requested feedback on ways to classify levels of domestication within a registry of domesticated applications:

https://spaces.at.internet2.edu/display/COmanage/Registry+of+Domesticated+Applications

The initial idea was to define 6 different levels of domestication:

Level 5 - planning stages only or not useful = blackbox, internal, not provisionable
Level 4 - purely internal = internal, provisionable
Level 3 - LDAP, not federated
Level 2 - Not standards compliant but offers a local aspect or hook that could be federated; community contribution plug-in = external, unsupported
Level 1a - Out of the box works with standard federated technologies, needs simple configuration without writing code = external, supported, federated
Level 1b - external, supported, some additional work needed

Danno commented that some of the info in the levels does not lend itself to scaling. For example, clearly Level 5 is less domesticatable than Level 1 . But between levels 3 and 2, it is not clear which is more domesticatable.

Heather stated that the idea of numbers was to add up numbers to come up with an metric, where the lower the number the better. A solution could be to use more of the 1a and 1b approach when it's not clear that one domestication level is better than another.

Danno remarked that it's important to remember that when a new version of a domesticated application comes out, it may not longer be domesticatable. For example, when there is a connector or plug-in that doesn't come from the vendor or open source project, it might not work when a new version is released.

It was suggested that maybe the best categories are

- Fully domesticatable
- Partially domesticatable
- Not at all domesticatable as currently instantiated

Bob stated that "built-in support for SAML" is not necessarily the ideal. The best scenario is for applications to have hooks so developers can do what they want externally. Benn noted that some organizations may not want to customize, but may be looking for an out-of-the-box solution.

Benn suggested there are three categories for domestication, and it's useful to distinguish between the three:
1. There is the ability to write "glue code" for domestication and/or this is required, because all that's provided by the application is hooks
2. There is some 3rd party support for domestication, could be open source or from a vendor
3. The application has vendor supported domestication

Heather noted that LIGO is interested in knowing how much code they will need to write to achieve domestication for an application.

After the call, the wiki page was updated to show these levels of domestication:

Level 0 - Not externalizable, not provisionable via automated tools (black box)
• Not considered domesticated.

Level 1 - Not externalizable, but provisionable via automated tools
• Not considered domesticated. e.g. for authn, this would imply passwords stored locally.

Level 2 - Externalizable via hooks
• Considered domesticatable.

Level 3 - Externalizable via third-party plugin
• Considered domesticated for available plugins.

Level 4 - Externalizable via vendor supported functionality
• Considered domesticated for supported functionality.

Heather encouraged others to contribute ideas on the wiki.

Next Call: Friday, August 6 at 2pm ET