UMA telecon 2016-02-04

UMA telecon 2016-02-04

Date and Time

- Thu Feb 4, 9-10am PT
  - Voice: Skype: +9905100000481 or US +1-805-309-2350 (international dial-in lines), room code 178-2540#
  - Screen sharing: http://join.me/findthomas - NOTE: IGNORE the join.me dial-in line shown here in favor of the dial-in info above (Kantara "line C" and the Skype line)
  - UMA calendar: http://kantarainitiative.org/confluence/display/uma/Calendar

Agenda

- (Note that a short ad hoc meeting to discuss issue #239 will be held just prior, on the same dial-in)
- Roll call
- Approve minutes of UMA telecon 2016-01-07
- Force-rank use cases in our UMA Roadmap for 2016 page: #IoT, #APIsec, #fedauthz, #RSctrl, #ROctrl, #wideeco, #trust, #security
- Appoint a small contingent to look at revising our charter
- Dig into the top item on our use case pile
- AOB

Minutes

Roll call

Quorum was not reaching.

Approve minutes

Deferred.

Force-rank use cases

What's A and what's B in terms of priority (since we probably can't get more "rank-y" than that)? John W suggests working from "likelihood of error occurring" and "impact of the error". The other factor is "detectability of the error". If the exploit is 100% undetectable, that's bad. This is a good way to prioritize #security work (which is overtly about bugs). For feature work prioritization, we had talked about gathering "customer need" in 2016 or 2017. Mike adds that this could include "developer adoption" – so are we targeting developers and deployers both? "Customers" would typically mean those who know they need UMA. "Developer adoption" is about building community among those who may not know they "need UMA" yet. (This is what Adrian's goal is around an always-on AS, to entice RS and client developers.) George questions that UMA should be looking for adoption in the OAuth fashion.

How do we define #wideeco? If we correctly define mechanisms for dynamic onboarding of disparate parties, wouldn't it be the case that the narrower ecosystems would tend to use those mechanisms to build their deployments? The question is whether dynamic mechanisms add too much overhead for static partnerships to be willing to use. So maybe it's not just about correct definition of mechanisms; it's about a continuum of needs. There's definitely a tension present around when to optimize for #wideeco, and what our options are. It's not just about how to gather claims securely from Bob when his claims providers and Alice's AS-as-a-claims-client have not pre-established trust between them. There are other considerations as well, as we look at the broader implications of what "value-add features" are desired.

At: Eve and James: Share technical paper/thoughts on potential solutions to "Bob when his claims providers and Alice's AS-as-a-claims-client have not pre-established trust between them"

So far – all still being discussed:

#IoT (Eve), #APIsec, #fedauthz, #RSctrl, #ROctrl, #wideeco (Adrian, Justin), #trust, #security (Justin, Eve)

Justin adds a vote for the issues he submitted that have to do with implementation simplification and, in some cases, making it an OAuth extension grant (?). This needs a new "hashtag".

At: Eve: Add a #simplification label and add the self-contained token discussion to that issue.

John W notes that constraining sharing based on geolocation would be a good idea. Eve suggests adding an issue on this.

At: Eve: Set up one more round of ad hoc #239 meetings next week, in hopes of finishing then.

Validation of self-contained token discussion on that back channel:

- Mike Schwartz@All: I don't think self-contained tokens is limited to IOT...
- Mike Schwartz@All: And also, I'm not sure its possible...
- Justin Richer@All: Self contained tokens are in no way limited to IoT and that's not hte only issue of IoT. And how is it not possible? People have been doing it for many, many years.
- George Fletcher@All: agreed that self-contained tokens are limited to IoT. I also think that while we should use JWT for the token syntax there are some real issues to solve here.
- Mike Schwartz@All: of course self contained tokens are possible, but the way we defined the RPT flow
- Mike Schwartz@All: the RPT gets issued, and its empty
• Mike Schwartz@All: and then permissions get added.
• Mike Schwartz@All: So its not that RPT can't be self-contained, but it can't be used as an access token
• Mike Schwartz@All: because the access token value would change
• Justin Richer@All: That's not how we issue RPTs. We don't issue RPTs until there are permissions to it. If we add permissions we'd issue a new RPT which is valid.
• George Fletcher@All: Yes, or become a "super" token which has it's own security issues.
• Justin Richer@All: Right, super tokens are a big problem in wider cases.

Charter revision

Deferred.

Use case work

See the issue #239 meeting series email thread for the continuing notes on that.

Attendees

As of 20 Jan 2016, quorum is 7 of 12. (Evariste, François, Domenico, Kathleen, Sal, Thomas, Andi, Robert, Maciej, Eve, Søren, Mike)

1. Eve
2. Mike
3. Kathleen
4. Andi

Non-voting participants:

• Justin
• John W
• George
• Adrian
• Sarah
• Jin