UMA telecon 2017-08-03

UMA telecon 2017-08-03

Date and Time

- Thursdays, 9-10am PT
- Screenshare and dial-in: http://join.me/findthomas
- UMA calendar: http://kantarainitiative.org/confluence/display/uma/Calendar

Agenda

- Roll call
- Approve minutes of UMA telecon 2017-07-20, UMA telecon 2017-07-27
- Schedule for upcoming weeks
  - Can’t meet Aug 10: substitute another time?
  - When are people on vacation?
- UMA V2.0 work:
  - Disposition of Comments/GitHub issues for V2.0/Grant swimlane/FedAuthz swimlane/Release Notes/UIG/Wikipedia/Grant rev 06/FedAuthz rev 06
  - We need to work through the open issues, so please weigh in on GitHub if you have opinions
  - Candidate motion when we’re ready: "Approve UMA 2.0 Grant rev nn and FedAuthz rev nn as amended by the instructions of UMA telecon 2017-xx-xx as Draft Recommendations and forward them to the Kantara Leadership Council to request immediate certification for an All-Member Ballot as Recommendations."
- UMA2 logo discussion (defer till we're in All-Member Ballot)
- AOB

Minutes

Roll call

Quorum was reached.

Approve minutes

Approve minutes of UMA telecon 2017-07-20, UMA telecon 2017-07-27: APPROVED by acclamation.

Schedule for upcoming weeks

Eve can’t make the Aug 10 meeting. Andi is away next week and the third week out. Domenico is out starting Aug 14. If we had an e-ballot running during the last week of August to approve the specs for a new Public Comment period, voting participants on the call could be available online the week of Aug 28 and the week of Sep 4 (that day itself is US Labor Day).

At: Eve: Send invitation for Monday Aug 7 at 9am PT as a substitute for Thursday Aug 10. Eve should be available at our normal times for the following three weeks; we’ll see if we can continue to get critical mass then (if not official quorum).

UMA 2.0 work

#337: This harks back to Pedro’s comments and Justin’s responses about the grant’s “agnosticism” to clients. The assumption wording in Grant Sec 3.3 now is: “The client has obtained OAuth client credentials from the authorization server, either dynamically through [RFC7591] or [OIDCDynClientReg], or alternatively through a static process, and is prepared to authenticate itself to the token endpoint if appropriate.” Instead we could say: “The client has obtained a client ID or a full set of client credentials as appropriate, either statically or dynamically (for example, through [RFC7591] or [OIDCDynClientReg]). This grant works with clients of both confidential and public types.” We have consensus.

#339: James misinterpreted the wording in FedAuthz Sec 4.1 and so suggested that the RS should get flexibility: object or array at its discretion for a single permission in the request, with the AS picking up the slack. The spec would have to clarify/change the wording and probably add an example of the single-object-in-an-array way of doing things. Reactions:

- Removing the object option would have been breaking. This isn’t.
- Typically, adding “two ways of doing things” would be unattractive (except for backwards compatibility reasons; we did that in V1.0.1 and then, in some cases, removed the old deprecated methods in V2.0).
- Any insights from OIDC history? Spec design philosophy has been to make the AS do the heavy lifting – we have exactly the same.

#341: Eve’s suggestion in the issue thread was that there are two use cases: synchronous and asynchronous. But what does that mean? She meant that by the time the RO responds, it would be impossible to consider their response during the RqP’s session. But the discussion from issue #289 is about how it’s not a matter of time elapsed, but use cases where the RO’s policy says “ask me every time”; such that not issuing a ticket at all means that the AS would have “amnesia” when the client returns even after a very lengthy period of time. If it returns directly to the AS with the ticket after, say, several days, ticket in tow, even if all the other necessary claims have expired etc., there might then be a record of the RO having said “yes” and claims-gathering could proceed as required. If we decide to change from MUST return a ticket to MAY, then all the client can do when the AS doesn’t return a ticket is obviously to start over at the RS. It turns into a terminal type of error from a continuance type of error. Ultimately, a MUST forces it to be one kind of error, while a MAY lets the error function for two different kinds of use cases. So we have two clear options (probably still adding that the AS should document its type of request submissions regardless?):
• Option 1: Keep the MUST, which is a terminal error.
• Option 2: Change to MAY, which splits into terminal and continuance errors.

UMA logo
Deferred.

Attendees
As of 7 Mar 2017, quorum is 4 of 7. (Domenico, Sal, Andi, Maciej, Eve, Mike, Cigdem)

1. Domenico
2. Sal
3. Andi
4. Maciej
5. Eve
6. Cigdem

Non-voting participants:
• Kathleen
• Bjorn

Regrets:
• James
• John W