UMA telecon 2016-09-22

UMA telecon 2016-09-22

Date and Time

- Thursdays, 9-10am PT
  - Skype: +9905100000481 / US +1-805-309-2350 / international lines / web calling interface / code 1782540
  - Screen sharing: http://join.me/findthomas - NOTE: do not use the join.me dial-in line
  - UMA calendar: http://kantarainitiative.org/confluence/display/uma/Calendar

Agenda

- Roll call
- Approve minutes of UMA telecon 2016-09-01, 2016-09-09
- Work on UMA.next issues
  - Core is up to 01
  - RSR is up to 00
  - Persisted claims token language proposal
  - Set math
- AOB

Minutes

Roll call

Quorum was reached.

Approve minutes

Approve minutes of UMA telecon 2016-09-01, 2016-09-09: APPROVED by acclamation.

Persisted claims token discussion

Justin has essayed an implementation. MatchAllClaimsOnAnyPolicy can pass in a PCT, and takes a union of the claims that are in the ticket and the (optional) PCT. PCT creation is straightforward (though not quite working yet because of a particular upstream library limitation). This basically tests what has to change, and the delta is relatively minimal (after having yanked out AATs and switched to using the OAuth endpoints vs. the RPT endpoint).

The AAT required a requesting party to authorize usage of the authorization API, which came bundled with the ability to optimize future trust elevation processes. No local AS account was required, though this might be a common way to do things. It used OAuth, and thus the RqP was required to act as an OAuth resource owner – which is too identification/authentication focused. We are now considering breaking apart the abilities to:

1. Elevate trust in the requesting party in a variety of ways (not just, say, requiring uniquely identifying authentication, as most OAuth-based account management implementation requires) – which has been a goal of UMA all along with its concept of CBAC
2. Optimize future such processes
3. Enable authorization for such optimization

Since there’s “no OAuth to lean on” in talking about Bob’s authorization for his client's interactions with the AS authorization interface now, what would it mean if the client were to be able to push claims before he could authorize that? This could have happened in OAuth too, e.g. through implicit flows, and employment contracts that okayed SAML assertion flows and the like. Now, we’re sort of acknowledging that authorization has always been in the realm of heuristics, but we'll have to be more explicit than we had been previously that deployers need to ensure they're presenting RqPs with the right authorization choices at the right time. The AAT was required and early-bound, and went out of its way to use OAuth's front channel; now it's optional and later-bound, with more options as to timing and UX, and goes out of its way to use OAuth's back channel. We can perhaps get more specific in our instructions for the "UMA grant" (the client-AS interface) vs. the "authorization interface" (everything else). Also, when the RqP isn't an end-user but rather operating a client autonomously, we've found a more elegant way to accommodate both "legal persons" and "natural persons", as the former don't go around authenticating.

Domenico's new graphic captures these concepts nicely.

Spec instructions:

- Justin to propose PCT text by end of week.

Attendees

As of 31 Aug 2016, quorum is 6 of 10. (Domenico, Sal, Nagesh, Andi, Robert, Maciej, Eve, Jeffrey, Mike, Sarah)

1. Domenico
2. Sal
3. Nagesh
4. Andi
5. Robert  
6. Eve  
7. Mike  
8. Sarah

Non-voting participants:

- John W
- Andrew
- Scott F
- James
- Jin
- Justin
- Arlene
- Ishan