

Current LTI Integration

Authentication

Chris Colvard's LTI authentication walkthrough:

(11:39:22 PM) cjcolvar: the lti omniauth provider is configured here: https://github.com/avalonmediasystem/avalon/blob/develop/config/authentication_iti.yml

(11:39:42 PM) cjcolvar: with a hash of key/secret pairs which are valid

(11:40:07 PM) cjcolvar: this gets read in here: https://github.com/avalonmediasystem/avalon/blob/develop/config/initializers/authn_iti_providers.rb

(11:40:28 PM) cjcolvar: and passed to devise here: <https://github.com/avalonmediasystem/avalon/blob/develop/config/initializers/devise.rb#L205-L207>

(11:41:51 PM) cjcolvar: that makes a omniauth callback route: `/users/auth/iti/callback`

(11:42:46 PM) cjcolvar: which is used to setup the lti tool inside the LMS

(11:43:48 PM) cjcolvar: For Oncourse(sakai):

(11:43:48 PM) cjcolvar: Required Information

(11:43:48 PM) cjcolvar: *Remote Tool Url: <http://lancelot.dlib.indiana.edu/users/auth/iti/callback>

(11:43:48 PM) cjcolvar: *Remote Tool Key: `samplekey`

(11:43:48 PM) cjcolvar: *Remote Tool Secret: `samplesecret`

(11:45:00 PM) cjcolvar: when accessing that tool, it posts to that callback route with a big context hash: <https://gist.github.com/cjcolvar/68814e64fd23902d216a>

(11:45:29 PM) atomical: you enter that information under Module in Canvas

(11:46:55 PM) cjcolvar: This hits the OmniauthCallbacksController which calls the provider implementation to verify that the authentication succeeded and if so calls `User#find_for_iti`: <https://github.com/avalonmediasystem/avalon/blob/develop/app/models/user.rb#L46-L51>

(11:47:59 PM) cjcolvar: Now we have a user and we setup the `virtual_groups` for it by pulling the `context_id` out of the posted hash

(11:48:19 PM) mbklein: And are all of the differences between what I can accomplish as a local user and what my LTI login will let me do encapsulated in the Ability class?

(11:48:35 PM) cjcolvar: right now, no

(11:48:46 PM) cjcolvar: it is handled here: <https://github.com/avalonmediasystem/avalon/blob/develop/app/models/user.rb#L61-L67>

(11:48:59 PM) cjcolvar: if there are virtual groups, only return those

(11:49:07 PM) cjcolvar: for `User#groups`

(11:49:42 PM) cjcolvar: so you wouldn't have the `manager/administrator/group_manager` group when logging in through lti

(11:50:06 PM) cjcolvar: we could switch that over to ability logic instead

(11:51:21 PM) cjcolvar: the virtual groups need to be stuffed into the session because the modified user object doesn't stick around during the whole request: https://github.com/avalonmediasystem/avalon/blob/develop/app/controllers/users/omniauth_callbacks_controller.rb#L44

(11:51:52 PM) mbklein: cool

(11:52:04 PM) cjcolvar: which then gets used throughout the session: https://github.com/avalonmediasystem/avalon/blob/develop/app/controllers/application_controller.rb#L35-L44

(11:52:55 PM) cjcolvar: the user is redirected to the search page with the virtual group facet applied: https://github.com/avalonmediasystem/avalon/blob/develop/app/controllers/users/omniauth_callbacks_controller.rb#L51-L52

(11:53:01 PM) cjcolvar: that's it for now

(11:53:41 PM) cjcolvar: I think we might need some sort of "lti_session" flag in the session so we're not relying upon the existence of `virtual_groups`

(11:53:51 PM) cjcolvar: and lealelu is reworking the configuration part

(11:51:20 PM) pdinh: How does the "provider implementation verify that the authentication succeeded"

(11:55:12 PM) cjcolvar: pdinh: it checks the key/secret pair as well as timestamps I believe

(11:55:14 PM) cjcolvar: <https://github.com/xaviaracil/omniauth-iti/blob/master/lib/omniauth/strategies/iti.rb>

(11:56:13 PM) cjcolvar: https://github.com/instructure/ims-iti/blob/master/lib/ims/iti/request_validator.rb