Ubuntu Online Accounts: credentials management for application developers

David King, amigadave@amigadave.com

2nd February 2013 11:45–12:05 / FOSDEM
Outline

1. Background
   - Overview
   - Components
   - UI components

2. Demo

3. Application developer guide

4. Summary
Outline

1. Background
   - Overview
     - Components
     - UI components

2. Demo

3. Application developer guide

4. Summary
History of accounts-sso

- non-UI components of UOA
- part of MeeGo and used in the N9
- will(?) be used in Tizen, KDE integration in progress
- LGPL 2.1 license
- hosted on Google Code
Terminology

- account: given by a provider to a user, and allows access to services
- provider: gives out accounts to users, as well as allowing them to use the accounts with a variety of services
- service: hosted by a provider, and allows the user to perform some task
- service type: describes the primary functionality provided by a service, and is a useful way to group similar services
Outline

1. Background
   - Overview
   - Components
     - UI components

2. Demo

3. Application developer guide

4. Summary
libaccounts-glib

- C/GObject account database implementation
- C++/Qt bindings (libaccounts-qt)
- stores account settings, but no passwords/secrets
- no dependency on other UOA components
libsignon

- C/GObject and C++/Qt wrappers around signond
- forwards requests over D-Bus to signond
- thin wrapper over the (slightly ugly) D-Bus API
helper modules loaded by the Control Center panel to create, edit, reauthenticate or delete an account
client of both libaccounts and libsignon
simple API (especially for OAuth accounts)
XML formats for describing providers, services and service types
Outline

1. Background
   - Overview
   - Components
   - UI components

2. Demo

3. Application developer guide

4. Summary
signon-ui

- provides WebKit view for OAuth authentication process
- provides simple dialogues for password authentication
- integrates with system indicator to prompt user for reauthentication
gnome-control-center-center-signon

- applet module for GNOME Control Center (requires patched Control Center)
- also available as a standalone preferences application
- use libaccounts to enumerate the accounts, and calls out to account-specific plugins as necessary
UOA demo

Play demo video
Basic concepts

- no account management UI provided in the application; provide a dedicated account plugin and invoke the Control Center panel instead
- enumerate the available accounts and select ones relevant to the application
- fetch the credentials and use them as normal (application-specific)
Enumerating accounts

- create a new AgManager for the desired service type, to restrict the emitted signals to be related only to that service
- connect to the “enabled-event” handler on the manager
- fetch the currently-enabled services and process them
- ignore the services which are disabled or part of a disabled account
Logging in

- fetch the AgAuthData from the AgAccountService
- optionally add further authentication parameters
- pass the signon information to a new SignonAuthSession, together with a result callback
- process the reply in the callback and either use the authentication parameters or report the error
Registering an application with UOA

- add an XML .application manifest and install it to $(prefix)/share/accounts/applications
- list services and service types if you want a service-specific description
- use intltool to provide gettext translations
Invoking the Online Accounts panel

- call the `credentials-preferences` script, which will in turn call the Control Center or standalone preferences binary
- use the `application=application-id` argument to start with an application selected at startup
- use the `account-details=account-id` argument to start with an account selected at startup
Adding an application-specific options widget

- replaces the in-application account options
- subclass the ApplicationPlugin class from libaccount-plugin
- tied to the settings panel, so must be a GModule which exports a GtkWidget (but could use XEmbed with a GtkSocket)
Summary

- UOA simplifies credentials management for application developers

Outlook

- convenience library for simpler UOA consumption by applications
- application-specific OAuth API keys
Acknowledgements

- Alberto Mardeegan, UOA tech lead
- FOSDEM and devroom organisers
- you, for listening
Convenience library for applications

- single library for applications to consume, rather than several
- possibility of different backends for non-UOA credentials providers
- easier to test/mock for applications
Ubuntu contributor agreement

- required for Launchpad-hosted UOA projects
- grants certain rights to Canonical
Web credentials indicator

- used to report authentication failures to the user
- can be queried by applications over D-Bus