lizard$_{a}uth_{s}erver Documentation$

Release

August 30, 2016

Contents

TODO: Short one-paragraph intro.

CHAPTER 1

Code documentation

CHAPTER 2

Project documentation

Changelog of lizard-auth-server

2.1 (unreleased)

- Added custom object managers for Profile and Company.

2.0 (2016-07-07)

- Added JWT expiration time of 15 minutes.
- Added new V2 API in separate endpoints which uses JWT.
- Added new models for SSO refactoring.
- Put user creation signals handlers into seperate module.
- Some py3 changes.
- Renamed 'return_unauthenticated'.
- A user arriving at the SSO server after being redirected there can now use a "return_unauthenticated" URL attribute. If the user is already logged in on the SSO server, redirects are set up so that he will be logged in on the site he was redirected from.

  If he is not, then if return_unauthenticated is False (the default, and the old behaviour), then he will be forced to log in before being redirected back.

  If return_unauthenticated, redirect the user back without logging in (to lizard-auth-client's /sso/local_not_logged_in/ URL).

  This enables a "attempt to auto-login if possible, but don't require it" workflow that is sometimes helpful.
- Python 3 setup and test fixes.

1.7 (2016-06-14)

- Upgraded to Django 1.9.7.

1.6 (2016-02-11)

- Added support for JSON Web Tokens. [byrman]
- Fixed wrong variable in log message. [reinout]

1.5 (2015-11-27)

- Moved `.clean()` method from the UserProfile model to a form. M2M fields cannot be checked by a model's `.clean()` as it always looks at the existing, old, data. [reinout]

1.4 (2015-11-27)

- One and only one 3di billing role is allowed for users with access to the 3di portal. [reinout]
- Added check that 3di billing isn't enabled 'for all users' of an organisation. [reinout]
- Added link to edit a user's profile at the end of the registration steps. This assumes the lizard6-style manual enabling of users. The previous link was in an unusable place. [reinout]

1.3 (2015-11-16)

- Added role inheritance, mainly based on an idea by Remco. One portal's role can point at other portals' roles as "inheriting roles". The other way around, the original role then becomes those other roles' "base role".

  If an organisation has an organisation role pointing at the base role *and* an organisation role pointing at the inheriting role, that inheriting role is available to the user (provided he has access to one of those two organisation roles). [reinout]
- Beautified the main SSO page ("my profile") and made it more usable. Nicer list of organisations; "definition list" instead of "table" for the user profile data; all actions in one spot. [reinout]
- Added separate page for viewing your permissions (which means "organisation-role-mappings") per portal, linked from the main portal page.

  As staff member, you can see detailed debug information to troubleshoot permissions. You can also view other users' permission information, essential for getting permissions right. [reinout]
- OrganisationRole has a manager now that automatically sets `select_related()` to select roles, portals and organisations. Otherwise to have to add select_related in way too many places. (Uncovered by testing with the django debug toolbar). Same for Role. [reinout]
- Added `select_related` in several places to lower the amount of queries, especially in the admin. [reinout]

1.2 (2015-11-02)

- Increased the test coverage. [reinout]
- Fixed bug with `__unicode__` method on `UserProfile`. [reinout]

1.1.1 (2015-10-30)

- Re-release of 1.1, I accidentally made it on the branch. [reinout]

1.1 (2015-10-30)

- Internal change: sorting the imports with `bin/isort lizard_auth_server/*py` now (and thus with `.isort.cfg`). Note: the imports aren't grouped in the regular 3 "pep8" groups. This is an experiment inspired by Plone. [reinout]
- Huge translation update. Everything is marked as translatable. Models and fields now have translatable names. Translation is set up to use https://translations.lizard.net, with instructions in the `README.rst`. And... everything has been translated into Dutch. [reinout]
- Huge admin update for the changelist pages. Better sorting, more columns, more search, more filtering, more links to related objects. [reinout]
- Huge update for the object edit pages. Better order, better fields, editable yes/no, etcetera. **Most important change**: horizontal filtering for portals instead of a long ctrl-click-to-select-multiple list. Also added inlines for easy editing roles on portals and editing organisation roles on organisations. [reinout]

1.0 (2015-09-24)

- The parameter to redirect to a different domain is now called `domain` instead of `next`. `next` is already used by django itself and it interferes too much.

  The `next` parameter is still supported if it starts with `http` for temporary backwards compatibility. [reinout]

0.8 (2015-09-18)

- Showing all organizations for a user. [remco]

0.7 (2015-08-26)

- The "allowed domain" setting for a site can now include multiple space-separated patterns. [byrman]
- Upgraded the test setup so that coverage is now also reported. We're at 56%. [reinout]

0.6 (2015-07-14)

- New API endpoints: get_roles, get_user_organisation_roles.

0.5 (2015-04-17)

- Compatibility with django 1.6: uidb64 instead of uidb36 hashed user IDs in password reset form. Password reset was broken after our move to django 1.6.

See https://docs.djangoproject.com/en/1.6/releases/1.6/#django-contrib-auth-password-reset-uses-base-64-encoding-of-user-pk

0.4 (2015-01-12)

- Added support for login on custom domains.

0.3 (2014-11-19)

- Added an internal API call that returns all organisations, so that they can be added to clients before any user of that organisation has logged in (lizard_auth_client has a `synchronise_organisations()` function).

0.2.5 (2014-05-16)

- Bug fix: do not crash on profile-less users.

0.2.4 (2013-10-17)

- More convenient Django Admin screens.

0.2.3 (2013-10-08)

- Fix bug that caused lizard-auth-server to return non-distinct organisation_roles (issue3).

0.2.2 (2013-09-04)

- Fix bug that caused activation to fail (organisations not saved correctly).

0.2.1 (2013-09-03)

- Failed to check in a crucial change.

0.2 (2013-09-02)

- Bug fix: only pass organisation-roles belonging to the current portal

0.1 (2013-08-30)

- Initial project structure created with nensskel 1.30.dev0.
- First release of lizard-auth-server based on a heavily modified django-simple-sso.
- Roles, Organisations and related data are now part of lizard_auth_server.

- Information about the user's roles in organisation is passed from VerifyView, along with information about the user. This is ignored by old versions of lizard_auth_client but can be used by a new version to construct the same information at the Portal side.

Credits

- TODO started this library

CHAPTER 3

lizard-auth-server

Lizard auth server is build upon django-simple-sso.

It is installed as https://sso.lizard.net, see https://github.com/nens/sso/

3.1 Workflow

The workflow follows the django simple sso workflow.

3.2 Updating translations

Go to the `lizard_auth_server` subdirectory and run makemessages and upload the catalog to transifex:

```
$ cd lizard_auth_server
$ ../bin/django makemessages --all
$ cd ..
$ bin/transifex upload_catalog
```

Then update the NL translation on https://translations.lizard.net/projects/p/lizardsystem/resource/lizard-auth-server/ and afterwards fetch the latest translations:

```
$ bin/transifex fetch
```

Note: this also fetches af/vi/zh, but we don't translate into those languages currently. They're ignored in the `.gitignore` file.

3.3 Development with docker

The short version:

```
$ docker-compose build
$ docker-compose run web python3 bootstrap.py
$ docker-compose run web bin/buildout
$ docker-compose run web bin/django migrate
$ docker-compose up
```

The site will now run on http://localhost:5000

Running the tests:

```
$ docker-compose run web bin/test
```