
Cerberus Access Control

Release 0.2.3

Dec 11, 2017

Contents

1	Cerberus Access Control	1
1.1	License	1
1.2	Installation	1
1.3	Documentation	1
1.4	Development	1
2	Contributing	3
2.1	Bug reports	3
2.2	Documentation improvements	3
2.3	Feature requests and feedback	3
2.4	Development	4
3	Authors	7
4	Changelog	9
4.1	0.2.2 (2017-12-11)	9
4.2	0.2.1 (2017-11-08)	9
4.3	0.2.0 (2017-07-03)	9
4.4	0.1.7 (2017-04-19)	10
4.5	0.1.6 (2017-04-19)	10
4.6	0.1.5 (2017-04-18)	10
4.7	0.1.4 (2017-04-10)	10
4.8	0.1.3 (2017-04-04)	11
4.9	0.1.2 (2017-04-03)	11
4.10	0.1.1 (2017-02-21)	11
5	Indices and tables	13

Django Ontology Based Access Control Module

1.1 License

Software licensed under [ISC](#) license.

1.2 Installation

```
pip install django-cerberus-ac
```

1.3 Documentation

On [ReadTheDocs](#)

1.4 Development

To run all the tests: `tox`

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

2.1 Bug reports

When [reporting a bug](#) please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

2.2 Documentation improvements

Cerberus Access Control could always use more documentation, whether as part of the official Cerberus Access Control docs, in docstrings, or even on the web in blog posts, articles, and such.

2.3 Feature requests and feedback

The best way to send feedback is to file an issue at <https://github.com/Genida/django-cerberus-ac/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

2.4 Development

To set up *django-cerberus-ac* for local development:

1. Fork *django-cerberus-ac* (look for the “Fork” button).
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/django-cerberus-ac.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

4. When you’re done making changes, run all the tests with one command:

```
tox
```

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

2.4.1 Pull Request Guidelines

If you need some code review or feedback while you’re developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run `tox`)¹.
2. Update documentation when there’s new API, functionality etc.
3. Add a note to `CHANGELOG.rst` about the changes.
4. Add yourself to `AUTHORS.rst`.

¹ If you don’t have all the necessary python versions available locally you can rely on...

- **Travis:** it will run the tests for each change you add in the pull request. It will be slower though...

- **pyenv:**

```
# important libraries to compile Python
sudo apt install -y libssl-dev openssl zlib1g-dev sqlite3 libsqlite3-dev libbz2-dev bzip2

git clone https://github.com/pyenv/pyenv.git ~/.pyenv
export PATH="${HOME}/.pyenv/bin:${PATH}"
eval "$(pyenv init -)"

pyenv install 3.5.3
pyenv install 3.6.0 # etc.
pyenv global system 3.5.3 3.6.0
```


2.4.2 Tips

To run a subset of tests:

```
tox -e envname -- py.test -k test_myfeature
```

To run all the test environments in *parallel* (you need to pip install detox):

```
detox
```


CHAPTER 3

Authors

- Pierre Parrend
- Mihnea Gheorghiu
- Timothée Mazzucotelli
- Florent Colin
- Jean-Louis Mandel

4.1 0.2.2 (2017-12-11)

- Update to support django-app-settings 0.3.0.

4.2 0.2.1 (2017-11-08)

- Fix `save()` prohibited to prevent data loss due to unsaved related object 'reference' error in `admin save_model`.

4.3 0.2.0 (2017-07-03)

- Override `save_models` admin methods to add user in history records.
- Add and rename methods in `RoleMixin`.
- Fix mapping setting check function.
- Implement implicit authorization.
- Add Science/Research classifier.
- Fix bug when id cannot be coerced to right type.
- Add migration 2.
- Improve `RolePrivilege` admin.
- Add drag and zoom to role hierarchy graph.
- Update initial migration to reflect code.

4.4 0.1.7 (2017-04-19)

- Add `d3.min.js` as a static asset to enable it in SSL context.

4.5 0.1.6 (2017-04-19)

- Finish implementation of edit privileges page (efficient below 1000*1000 matrices).

4.6 0.1.5 (2017-04-18)

- Improve admin display.
- Add `allow_update_own_privileges` setting.
- Improve `instance_from_method`.
- Add `skip_implicit` and `log` kwargs on `can` method.
- Fix role hierarchy chart, use `allow_own_update` setting.
- Put back `setting_prefix`.
- Add `django-app-settings` dependency, remove `autowidth` from `data-table`.
- Move access to objects into `get` to avoid migrate errors.
- Remove Github pages configuration file.
- Update django from 1.10.6 to 1.11.
- Fix migrations (missing `__init__.py`).

4.7 0.1.4 (2017-04-10)

- Implement Ajax call with notification.
- Interface `no_datatable` template with backend.
- Set namespace to `cerberus`, improve consistency.
- Match application name in breadcrumbs.
- Move application settings into `apps` module.
- Change IDs from `int` to `str`.
- Add migration file, avoid loading role instances at start-up.
- Change mapping setting.
- Register models in admin.
- Fix unique constraint with integers default 0.

4.8 0.1.3 (2017-04-04)

- Admin classes added for separation of privileges.
- Models updated.

4.9 0.1.2 (2017-04-03)

- Added views, templates, and models for testing.
- Changed Bootstrap version from 3.3.1 to 2.3.1 and JQuery 3.1.1 to 2.1.1.

4.10 0.1.1 (2017-02-21)

- Alpha release on PyPI.

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`