
repoze.who SA Documentation

Release 1.0rc2

Gustavo Narea

April 05, 2011

CONTENTS

1	How to install	3
2	Authenticator	5
3	Metadata provider	7
4	Miscellaneous	9
5	How to get help?	11
6	Contents	13
6.1	<code>repoze.who.plugins.sa</code> releases	13
7	Indices and tables	15
	Python Module Index	17
	Index	19

Module author: Gustavo Narea <me@gustavonarea.net>

Author Gustavo Narea.

Latest version 1.0rc2

Overview

The `repoze.who` SQLAlchemy plugin provides an authenticator and a metadata provider plugins for SQLAlchemy or Elixir-based models.

HOW TO INSTALL

The minimum requirements `repoze.who` and `SQLAlchemy` and you can install it all by running:

```
easy_install repoze.who.plugins.sa
```

The development mainline is available at the following Subversion repository:

```
http://svn.repoze.org/whoplugins/whoalchemy/trunk/
```


AUTHENTICATOR

```
class repoze.who.plugins.sa.SQLAlchemyAuthenticatorPlugin (user_class,  
                                                           db-  
                                                           ses-  
                                                           sion)
```

repoze.who authenticator for SQLAlchemy models.

Example:

```
from repoze.who.plugins.sa import SQLAlchemyAuthenticatorPlugin  
from yourcoolproject.model import User, DBSession
```

```
authenticator = SQLAlchemyAuthenticatorPlugin(User, DBSession)
```

This plugin assumes that the user name is kept in the `user_name` attribute of the users' class, as well as that such a class has a method that verifies the user's password against the password provided through the login form (it receives the password to be verified as the only argument and such method is assumed to be called `validate_password`).

If you don't want to call the attributes above as `user_name` and/or `validate_password`, respectively, then you have to "translate" them as in the sample below:

```
# You have User.username instead of User.user_name:  
authenticator.translations['user_name'] = 'username'
```

```
# You have User.verify_password instead of User.validate_password:  
authenticator.translations['validate_password'] = 'verify_password'
```

Note: If you want to configure this authenticator from an ini file, use `make_sa_authenticator()`.

```
repoze.who.plugins.sa.make_sa_authenticator (user_class=None,  
                                              dbsession=None,  
                                              user_name_translation=None,  
                                              vali-  
                                              date_password_translation=None)
```

Configure `SQLAlchemyAuthenticatorPlugin`.

Parameters

- **user_class** (*str*) – The SQLAlchemy/Elixir class for the users.
- **dbsession** (*str*) – The SQLAlchemy/Elixir session.
- **user_name_translation** (*str*) – The translation for user_name, if any.
- **validate_password_translation** (*str*) – The translation for validate_password, if any.

Returns The authenticator.

Return type SQLAlchemyAuthenticatorPlugin

Example from an *.ini file:

```
# ...
[plugin:sa_auth]
use = repoze.who.plugins.sa:make_sa_authenticator
user_class = yourcoolproject.model:User
dbsession = yourcoolproject.model:DBSession
# ...
```

Or, if you need translations:

```
# ...
[plugin:sa_auth]
use = repoze.who.plugins.sa:make_sa_authenticator
user_class = yourcoolproject.model:User
dbsession = yourcoolproject.model:DBSession
user_name_translation = username
validate_password_translation = verify_password
# ...
```

METADATA PROVIDER

class `repoze.who.plugins.sa.SQLAlchemyUserMDPlugin` (*user_class, db-session*)
repoze.who metadata provider that loads the SQLAlchemy-powered object for the current user.

It loads the object into `identity['user']`.

Example:

```
from repoze.who.plugins.sa import SQLAlchemyUserMDPlugin
from yourcoolproject.model import User, DBSession
```

```
mdprovider = SQLAlchemyUserMDPlugin(User, DBSession)
```

This plugin assumes that the user name is kept in the `user_name` attribute of the users' class. If you don't want to call the attribute above as `user_name`, then you have to "translate" it as in the sample below:

```
# You have User.username instead of User.user_name:
mdprovider.translations['user_name'] = 'username'
```

Note: If you want to configure this plugin from an ini file, use `make_sa_user_mdprovider()`.

```
repoze.who.plugins.sa.make_sa_user_mdprovider (user_class=None,
                                              dbsession=None,
                                              user_name_translation=None)
```

Configure `SQLAlchemyUserMDPlugin`.

Parameters

- **user_class** (*str*) – The SQLAlchemy/Elixir class for the users.
- **dbsession** (*str*) – The SQLAlchemy/Elixir session.
- **user_name_translation** (*str*) – The translation for `user_name`, if any.

Returns The metadata provider.

Return type SQLAlchemyUserMDPlugin

Example from an *.ini file:

```
# ...
[plugin:sa_md]
use = repoze.who.plugins.sa:make_sa_user_mdprovider
user_class = yourcoolproject.model:User
dbsession = yourcoolproject.model:DBSession
# ...
```

Or, if you need translations:

```
# ...
[plugin:sa_md]
use = repoze.who.plugins.sa:make_sa_user_mdprovider
user_class = yourcoolproject.model:User
dbsession = yourcoolproject.model:DBSession
user_name_translation = username
# ...
```

MISCELLANEOUS

`class repoze.who.plugins.sa.SQLAlchemyUserChecker` (*user_class, db-session*)
User existence checker for `repoze.who.plugins.auth_tkt.AuthTktCookiePlugin`.

Example:

```
from repoze.who.plugins.sa import SQLAlchemyUserChecker
from yourcoolproject.model import User, DBSession

checker = SQLAlchemyUserChecker(User, DBSession)
```

This plugin assumes that the user name is kept in the `user_name` attribute of the users' class. If you don't want to call it that way, then you have to "translate" it as in the sample below:

```
# You have User.username instead of User.user_name:
checker.translations['user_name'] = 'username'
```


HOW TO GET HELP?

The preferred place to ask questions is the [Repoze mailing list](#) or the #repoze IRC channel. Bugs reports and feature requests should be sent to the [issue tracker](#) of the [Repoze project](#).

CONTENTS

6.1 `repoze.who.plugins.sa` releases

This document describes the releases of `repoze.who.plugins.sa`.

6.1.1 `repoze.who.plugins.sa 1.0rc2` (2009-06-27)

- Added `repoze.who.plugins.sa.SQLAlchemyUserChecker`, a user checker for `repoze.who.plugins.auth_tkt.AuthTktCookiePlugin`.

6.1.2 `repoze.who.plugins.sa 1.0rc1` (2009-01-26)

- Introduced the `repoze.who.plugins.sa.SQLAlchemyUserMDPlugin` meta-data provider.
- Minor docstring fixes.

6.1.3 `repoze.who.plugins.sa 1.0b3` (2009-01-08)

Fixed [Bug #56](#) (`User.user_name` was not translatable).

6.1.4 `repoze.who.plugins.sa 1.0b2` (2008-12-18)

Renamed `repoze.who.plugins.sqlalchemy` to `repoze.who.plugins.sa` due to problems with the namespace.

6.1.5 `repoze.who.plugins.sqlalchemy 1.0b1` (2008-12-18)

Initial release.

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

PYTHON MODULE INDEX

r

`repoze.who.plugins.sa`, [3](#)

INDEX

M

`make_sa_authenticator()` (in module `repoze.who.plugins.sa`), [5](#)
`make_sa_user_mdprovider()` (in module `repoze.who.plugins.sa`), [7](#)

R

`repoze.who.plugins.sa` (module), [1](#)

S

`SQLAlchemyAuthenticatorPlugin` (class in `repoze.who.plugins.sa`), [5](#)
`SQLAlchemyUserChecker` (class in `repoze.who.plugins.sa`), [9](#)
`SQLAlchemyUserMDPlugin` (class in `repoze.who.plugins.sa`), [7](#)