django-password-reset Documentation

Release 0.9

Bruno Renié

June 01, 2016

Contents

1 Quickstart **3**
- 1.1 Installation . . . 3
- 1.2 Usage . . . 3
- 1.3 What you get . . . 3
- 1.4 What you can do . . . 3

2 Views **5**
- 2.1 Recover . . . 5
- 2.2 RecoverDone . . . 6
- 2.3 Reset . . . 6
- 2.4 ResetDone . . . 6

3 Signals **7**
- 3.1 `password_reset.signals.user_recovers_password` . . . 7

4 Changelog **9**

5 Indices and tables **11**

Class-based views for password reset, the usual "forget password?" workflow:

- User fills his email address or username
- Django sends him an email with a token to reset his password
- User chooses a new password

The token is not stored server-side, it is generated using Django's signing functionality.

- Author: Bruno Renié and contributors
- Licence: BSD
- Compatibility: Django 1.4+ (cryptographic signing needed)

Contents:

CHAPTER 1

Quickstart

1.1 Installation

Simple:

- `pip install django-password-reset`

1.2 Usage

Simple:

- Add `password_reset` to your `INSTALLED_APPS`
- `include('password_reset.urls')` in your root `urls.py`
- Link to the password reset page: `{% url "password_reset_recover" %}`
- Create a `password_reset/base.html` template and adapt it to your site's structure

1.3 What you get

- A *password reset* workflow with no data stored on the server, tokens are signed and checked with your `SECRET_KEY`.
- The ability to look for your user's username or email address.
- Password reset links that expire in two days (configurable).

1.4 What you can do

- Use custom templates for everything: the email subject and body, the forms and confirmation pages.
- Use custom forms if you need something else than searching for username *or* email, or search case-insensitively.
- Use a custom salt or expiration time for tokens (expiration via `PASSWORD_RESET_TOKEN_EXPIRES` setting).
- Allow password recovery for all users (default) or only for active users (via `RECOVER_ONLY_ACTIVE_USERS=False` setting)

See the next section.

CHAPTER 2

Views

Note: The `Recover` and `Reset` views share a common attribute, `salt`. This is the salt used for signing the password reset link, it is useful for avoiding collisions with values you may have signed elsewhere in your app. It doesn't need to be complicated, just distinct from other salts you're using in your code. More importantly, the salt must be the same on the `Recover` and the `Reset` views. The default salt is `password_recovery`. If you're not already using this as a salt somewhere else in your app, you don't need to alter it.

Additionally, there is a `url_salt` used for redirecting the user after he has entered his username or email. This salt **must** be different than the other one. Its default value is `password_recovery_url`.

2.1 Recover

This is a `FormView` that asks for a username or email, finds the corresponding user object and sends him an email.

2.1.1 Attributes

- `case_sensitive`: whether to search case-sensitively based on the form data. Default: `True`.
- `form_class`: the form to use for validating the user. Default: `password_reset.forms.PasswordRecoveryForm`. To customize form error messages, subclass the form and override the `error_messages` attribute.
- `success_url_name`: the name of the URL to redirect to after sending the recovery email. Change it if you don't use the provided URLconf. Defaults to `password_reset_sent`.
- `template_name`: defaults to `password_reset/recovery_form.html`.
- `email_template_name`: the template to use for sending the reset link by email. Default: `password_reset/recovery_email.txt`.
- `email_subject_template_name`: the template to use for generating the email subject. Defaults to `password_reset/recovery_email_subject.txt`.
- `search_fields`: the fields to search for on the `User` model. Default is `['username', 'email']`, you can restrict it to `['username']` or `['email']` but no other fields are supported, at least not with the default form class.

2.1.2 Methods

- `send_notification()`: this builds the email context, loads the template and sends the password reset email.
- `get_site()`: method to obtain the website's host name. This method is simply a wrapper around Django's get_current_site.

2.2 RecoverDone

This is a `TemplateView` to confirm the user that an email was sent.

2.2.1 Attributes

- `template_name`: `password_reset/reset_sent.html`

2.2.2 Template Context

`invalid` Set to `True` if the URL signature isn't valid, which happens if you change your `SECRET_KEY`, the `url_salt` or if people try to reverse-engineer your URLs.

`email`: the username or email of the user.

`timestamp`: the time the signature was issues, which normally corresponds to the time the reset email was sent.

2.3 Reset

2.3.1 Attributes

- `form_class`: defaults to `password_reset.forms.PasswordResetForm`. To customize form error messages, subclass the form and override the `error_messages` attribute.
- `token_expires`: expiration time (in seconds) of the password reset token. Default is two days.
- `template_name`: defaults to `password_reset/reset.html`.
- `success_url`: the URL to redirect to after a successful password reset. Defaults to `reverse_lazy('password_reset_done')`, change it if you don't use the provided URLconf.

2.3.2 Methods

- `invalid()`: this method builds the response returned when an invalid token is encountered.

2.4 ResetDone

This is a simple `TemplateView` that displays a success message. Its default `template_name` is `password_reset/recovery_done.html`.

CHAPTER 3

Signals

3.1 password_reset.signals.user_recovers_password

This signal is sent after a user successfully recovers their password. It provides the `user` instance as well as the `request` object from the view.

CHAPTER 4

Changelog

- 0.9 (2016-06-01):
  - Allow token expiration time to be customized with a setting.
- 0.8.2 (2016-01-12):
  - Django 1.9 compatibility (Josh Kelley).
- 0.8.1 (2015-10-30):
  - Add pt_BR translation (GitHub user eduardo-matos).
- 0.8 (2015-10-30):
  - Allow customizing form error message via the `error_messages` attribute on form classes.
  - Add Georgian translation (GitHub user gigovich).
  - Add Norwegian translation (GitHub user gunnaringe).
  - Tested on django 1.5 to 1.8 and Python 2.6 to 3.4.
- 0.7 (2014-02-18):
  - Return user instance in `PasswordResetForm.save()`, add `commit` keyword argument.
- 0.6.1 (2014-02-14):
  - Fix for custom user models without any field named `username`. Properly take `USERNAME_FIELD` into account.
  - Add German translation (GitHub user billyBlaze).
  - Add Chinese translation (GitHub user saggit).
- 0.6 (2013-12-15):
  - New `user_recovers_password` signal (José Sazo).
- 0.5.1 (2013-10-31):
  - Spanish, Polish and Russian translations.
- 0.5 (2013-05-19):
  - Support for Django 1.5's custom user model.
- 0.4 (2013-02-18):
  - Python3 and Django 1.5 support.
- 0.3:

  - The recover view now redirects to a signed URL to avoid duplicate submissions.
  - Bugfix: made `case_sensitive` work properly when set to `False`.
- 0.2: Bugfix: actually save the new password.
- 0.1: Initial version.

CHAPTER 5

Indices and tables

- genindex
- modindex
- search