Django Admin View Permission Documentation

Release 0.1

Context Information Security

Mar 19, 2018

Contents

1 Installation **3**
- 1.1 Getting the code . . . 3
- 1.2 Requirements . . . 3
- 1.3 Support . . . 3
- 1.4 Setup . . . 3

2 Configuration **5**
- 2.1 ADMIN_VIEW_PERMISSION_MODELS . . . 5

3 Management commands **7**
- 3.1 fix_proxy_permissions . . . 7

4 Uninstall **9**

Contents:

CHAPTER 1

Installation

1.1 Getting the code

The recommended way to install the `Admin View Permission` is via pip:

```
$ pip install django-admin-view-permission
```

To test an upcoming release, you can install the in-development version instead with the following command:

```
$ pip install -e git+https://github.com/django-admin-view-permission/django-admin-
↪view-permission.git#egg=django-admin-view-permission
```

1.2 Requirements

- Django

1.3 Support

- Django: 1.8, 1.9, 1.10, 1.11, 2.0
- Python: 2.7, 3.4, 3.5, 3.6

1.4 Setup

Make sure that `'django.contrib.admin'` is set up properly and add `'admin_view_permission'` to your `INSTALLED_APPS` setting:

```
INSTALLED_APPS = [
    'admin_view_permission',
    # ...
    'django.contrib.admin',
    # ...
]
```

Finally, run `python manage.py migrate` to create the view permissions.

CHAPTER 2

Configuration

The admin view permission provides one setting that you can add in your project's settings module to customize its behavior.

2.1 ADMIN_VIEW_PERMISSION_MODELS

This setting defines which models you want to be added the view permission. If you don't specify this setting then the view permission will be applied to all the models.

2.1.1 Example

```
ADMIN_VIEW_PERMISSION_MODELS = [
    'auth.User',
    ...
]
```

CHAPTER 3

Management commands

The admin view permission provides one management command which fixes the permissions on the proxy models.

3.1 fix_proxy_permissions

This command will create the appropriate entries on the *ContentType* and *Permission* models. Then it will delete the permissions, which are created from django *migrate* command and are associated with the parent model. More information you can find here.

3.1.1 Example

```
python manage.py fix_proxy_permissions
```

CHAPTER 4

Uninstall

To remove the application completely firstly remove the `admin_view_permission` from your `INSTALLED_APPS` setting and then open a debug shell and execute the following commands in order to remove these extra permissions from the database:

```
from django.contrib.auth.models import Permission
permissions = Permission.objects.filter(codename__startswith='view')
permissions.delete()
```

Note: Before delete the permission would be helpful to check if the permissions queryset contains only the view permissions and not anything else.