
wagtail-2fa Documentation

Release 1.6.0

Lab Digital

Jun 10, 2022

Contents

1	Installation	3
2	Settings	5
3	Making 2FA optional	7
4	Sandbox	9

This Django app adds two factor authentication to Wagtail. Behind the scenes it use [django-otp](#) which supports Time-based One-Time Passwords (TOTP). This allows you to use various apps like Authy, Google Authenticator, or 1Password.

CHAPTER 1

Installation

```
pip install wagtail-2fa
```

Then add the following lines to the `INSTALLED_APPS` list in your Django settings:

```
INSTALLED_APPS = [  
    # ...  
    'wagtail_2fa',  
    'django_otp',  
    'django_otp.plugins.otp_totp',  
    # ...  
]
```

Next add the required middleware to the `MIDDLEWARE`. It should come after the `AuthenticationMiddleware`:

```
MIDDLEWARE = [  
    # .. other middleware  
    # 'django.contrib.auth.middleware.AuthenticationMiddleware',  
  
    'wagtail_2fa.middleware.VerifyUserMiddleware',  
  
    # 'wagtail.core.middleware.SiteMiddleware',  
    # .. other middleware  
]
```

Migrate your database:

```
python manage.py migrate
```


CHAPTER 2

Settings

The following settings are available (Set via your Django settings):

- `WAGTAIL_2FA_REQUIRED` (default `False`): When set to `True` all staff, superuser and other users with access to the Wagtail Admin site are forced to login using two factor authentication.
- `WAGTAIL_2FA_OTP_TOTP_NAME` (default `False`): The issuer name to identify which site is which in your authenticator app. If not set and `WAGTAIL_SITE_NAME` is defined it uses this. sets `OTP_TOTP_ISSUER` under the hood.

Making 2FA optional

With the default `VerifyUserMiddleware` middleware, 2FA is enabled for every user. To make 2FA optional, use the `VerifyUserPermissionsMiddleware` middleware instead.

To do so, use the `VerifyUserPermissionsMiddleware` middleware instead of the `VerifyUserMiddleware` in your Django settings:

```
MIDDLEWARE = [  
    # ...  
    'wagtail_2fa.middleware.VerifyUserMiddleware',  
    'wagtail_2fa.middleware.VerifyUserPermissionsMiddleware',  
    # ...  
]
```

When this middleware is used, a checkbox is added to the group permissions and 2FA can be enabled or disabled per group.

2FA is always enabled for superusers, regardless of the middleware used.

CHAPTER 4

Sandbox

First create a new virtualenv with Python 3.8 and activate it. Then run the following commands:

```
make sandbox
```

You can then visit <http://localhost:8000/admin/> and login with the following credentials:

- E-mail: `superuser@example.com`
- Password: `testing`