Yubico Django Authentication Backend Documentation

Release 0.2

Tomaz Muraus

Jul 07, 2017

Contents

Author Tomaz Muraus <tomazREMOVE@tomaz.me>

Maintainer Tomaz Muraus <tomazREMOVE@tomaz.me>

Version 0.2

Source github.org

Bug tracker http://github.com/issues

Yubico Django is an authentication backend for Django framework which supports Yubikey OTP authentication.

To get up and started quickly, visit the *Installation* page.

Contents:

Installation

Step 1: Install this module

You can use the following command to install this module from PyPi:

```
pip install django_yubico
```

Alternatively you can also install the latest development version from the git repository:

```
pip install -e git+https://github.com/Kami/django-yubico-authentication-backend
↪#egg=django-yubico
```

Step 2: Add django_yubico to the INSTALLED_APPS

Once the `django_yubico` is in your Python path, you need to modify the `INSTALLED_APPS` setting to include the `django_yubico` module:

```
INSTALLED_APPS = (
    # ...,
    # Third-party
    'django_yubico',
    # ...,
)
```
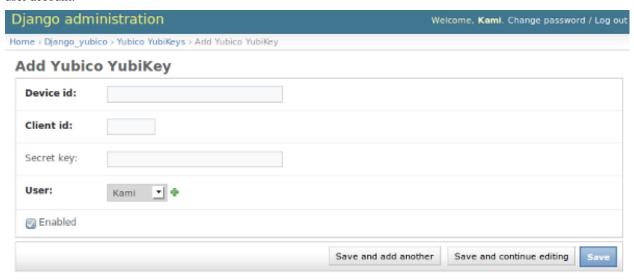
Step 3: Run syncdb to generate the necessary table

Run syncdb (this will create the database table which holds data about the YubiKeys):
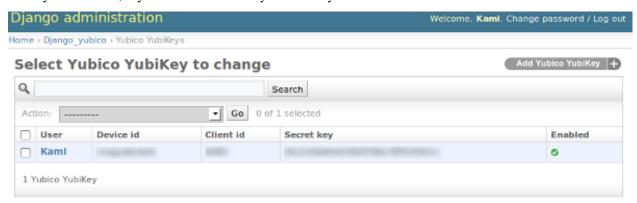
    python manage.py syncdb

Step 4: Login to the admin panel and add one or more YubiKeys

Login to the Django admin panel, visit the `Django_yubico` application setting and add a new YubiKey for your user account:

- `Device id` - the first 12 characters of the token (you can obtain it by generating an OTP and taking first 12 characters)

- `Client id` - your client id (you can obtain it by visiting the Yubico website)

- `Secret key` - this field is optional and you only need to specify it if you want the underlying client to verify the server response message HMAC-SHA1 signature (you can obtain it on the same page where you got your client id)

- `User` - The user which will be able to login with this YubiKey (remember that you can map a single YubiKey to multiple users)

- `Enabled` - You can optionally disable this YubiKey (meaning that you won't be able to login using this YubiKey until you enable it)

When you click save, key should be successfully added and you are almost done.

Step 5: Enable the custom authentication backend

To activate this backend you need at least put `django_yubico.backends.YubicoBackend` line to the `AUTHENTICATION_BACKENDS` tuple:

> AUTHENTICATION_BACKENDS = ('django_yubico.backends.YubicoBackend',
>
>)

For more information about the available backends and how they work, please visit the available authentication backends page.

Step 6: Load the module urls.py file

Put the following line in your `urls.py` file:

```
(r'^yubico/', include('django_yubico.urls')),
```

Step 7: Test if everything works

Visit http://yourpage.com/yubico/login/ and if everything went ok you should be able to login using your website username, OTP generated by YubiKey and a password.

By default you need to enter both - first your username and OTP and in the second step, your account password. For more information how to change this behavior, visit the *Settings* page.

Available authentication backends

This module offers the following three authentication backends.

YubicoBackend

This is a base backend which must be enabled if you want to use the YubiKey authentication.

You can enable it by putting the following lines to your `settings.py` file:

```
AUTHENTICATION_BACKENDS = (
        'django_yubico.backends.YubicoBackend',
    )
```

If you still want to allow other users without a YubiKey to log in, you must enable the `django.contrib.auth.backends.ModelBackend` as well:

```
AUTHENTICATION_BACKENDS = (
        'django_yubico.backends.YubicoBackend',
        'django.contrib.auth.backends.ModelBackend',
    )
```

YubicoBackendStaff

This backend should be used in combination with the `YubicoBackend` backend and requires all the staff and super users to use the YubiKey to log in (normal users with or without a YubiKey will still be able to log in using their password):

```
AUTHENTICATION_BACKENDS = (
            'django_yubico.backends.YubicoBackend',
            'django_yubico.backends.YubicoBackendStaff',
    )
```

YubicoBackendRequireYubikey

This backend should also be used in combination with the `YubicoBackend` backend and requires **all** the users with at least one **active / enabled** YubiKey to log in using the YubiKey:

```
AUTHENTICATION_BACKENDS = (
            'django_yubico.backends.YubicoBackend',
            'django_yubico.backends.YubicoBackendRequireYubikey',
    )
```

Settings

The following settings are available:

YUBICO_USE_PASSWORD

Defaults to true `True` and means that user will also need to enter his account password after entering the OTP. If you want to allow user to only use his YubiKey to login, set this to `False`.

YUBIKEY_PASSWORD_ATTEMPTS

Defaults to true `3` and means how many times user can enter a wrong password before he needs to provide a new OTP. This helps to prevent brute forces attacks when someone gets a valid token or steals user's session cookie.

Note that this setting only has an effect if `YUBICO_USE_PASSWORD` is set to `True`.

YUBIKEY_SESSION_USER

The name of the session key where the user object is saved. Defaults to `yubicodjango_user`.

YUBIKEY_ATTEMPT_COUNTER

The name of the session key which holds the value of how many times user has entered the wrong password. Defaults to `yubicodjango_counter`.

YUBICO_MULTI_MODE

Defaults to `False`.

If set to `True` user will need to enter `YUBICO_MULTI_NUMBER` number of OTPs which were generated in the `YUBICO_MULTI_TIMEOUT` seconds long time window for a successful validation.

YUBICO_MULTI_NUMBER

Defaults to 3.

The number of OTPs user will need to enter when multi mode is enabled.

Note: This setting is only applicable is YUBICO_MULTI_MODE is set to True.

YUBICO_MULTI_TIMEOUT

Defaults to 10.

How many seconds can pass between the time when the first and the last OTP is generated.

Note: This setting is only applicable is YUBICO_MULTI_MODE is set to True.

FAQ

Does this module work with Django 1.2?

Yes, this module is tested and should work fine with Django 1.2.

Does this module support offline authentication?

No, this module depends on the yubico-python module and only supports the safest, online OTP authentication against Yubico or your own validation servers.

Can multiple users use the same YubiKey to log in?

Yes, the only requirement is that your website account usernames are unique.

This is required because user must enter his username + OTP to log in.

If you want more users to share the same YubiKey, it would be the safest to enable the `YUBICO_USE_PASSWORD` setting (you can read more about the available settings at the *Settings* page).

How can I customize the login templates?

You can customize the login templates by copying the `login.html` and `password.html` files from the `django_yubico/templates/django_yubico/` folder to your Django application templates folder and editing them (you must preserve the directory structure or change the path to the template files in `django_yubico/views.py`).

Changelog

Version 0.2.dev (09.05.2010)

- Added multi-mode support
- Fixed a bug where YubicoBackend class in some cases on failed validation returned False instead of None

Version 0.1.dev (07.05.2010)

- Initial release

Other

Special thanks to RudolphFro, the author of the original yubikey-python module for the idea and initial implementation.

Links and references

Links which you may find useful if you want to learn more about the YubiKey and validation protocol.

- Yubico developers intro - introduction to the YubiKey for the developers
- Yubico Web Service API - YubiKey web service API documentation
- Validation Protocol Version 2.0 - validation protocol version 2.0 specifications and description
- Server v2 FAQ - frequently asked questions about the new validation protocol

CHAPTER 2

Indices and tables

- genindex
- modindex
- search