**SonataUserBundle**

https://sonata-project.org/bundles/user/master/doc/reference/installation.html

1. **Prérequis :**

pour installaer SonataUserBundle il faut installer :

* SonataAdminBundle(on a déja fait)
* SonataEasyExtendsBundle :

composer require sonata-project/easy-extends-bundle

Activer le bundle dans le fichier AppKernel.php :

new Sonata\EasyExtendsBundle\SonataEasyExtendsBundle(),

1. **Installation :**

composer require sonata-project/user-bundle --no-update

composer require sonata-project/doctrine-orm-admin-bundle --no-update # optional

composer require friendsofsymfony/rest-bundle --no-update # optional when using api

composer require nelmio/api-doc-bundle --no-update # optional when using api

composer require sonata-project/google-authenticator --no-update # optional

composer update

Activer les bundles :

**new** Sonata\EasyExtendsBundle\SonataEasyExtendsBundle(),

// ...

**new** FOS\UserBundle\FOSUserBundle(),

**new** Sonata\UserBundle\SonataUserBundle(),

1. **Configuration :**

* **Acl configuration :**

**#config.yml**

**sonata\_user:**

**security\_acl:** true

**manager\_type:** orm *# can be orm or mongodb*

*# security.yaml*

security:

*# [...]*

encoders:

FOS\UserBundle\Model\UserInterface: sha512

acl:

connection: default

* **DOCTRINE CONFIGURATION :**

*# config.yaml*

doctrine:

*#...*

dbal:

types:

json: Sonata\Doctrine\Types\JsonType

* **FOSUSERBUNDLE CONFIGURATION :**

*# config.yaml*

fos\_user:

db\_driver: orm *# can be orm or odm*

firewall\_name: main

user\_class: Sonata\UserBundle\Entity\BaseUser

group:

group\_class: Sonata\UserBundle\Entity\BaseGroup

group\_manager: sonata.user.orm.group\_manager *# If you're using doctrine orm (use sonata.user.mongodb.group\_manager for mongodb)*

service:

user\_manager: sonata.user.orm.user\_manager

from\_email:

address: "%mailer\_user%"

sender\_name: "%mailer\_user%"

* **INTEGRATING THE BUNDLE INTO THE SONATA ADMIN BUNDLE :**

*# config/routing.yaml*

sonata\_user\_admin\_security:

resource: '@SonataUserBundle/Resources/config/routing/admin\_security.xml'

prefix: /admin

sonata\_user\_admin\_resetting:

resource: '@SonataUserBundle/Resources/config/routing/admin\_resetting.xml'

prefix: /admin/resetting

*# config/security.yaml*

security:

firewalls:

*# Disabling the security for the web debug toolbar, the profiler and Assetic.*

dev:

pattern: ^/(\_(profiler|wdt)|css|images|js)/

security: false

*# -> custom firewall for the admin area of the URL*

admin:

pattern: /admin(.\*)

context: user

form\_login:

provider: fos\_userbundle

login\_path: /admin/login

use\_forward: false

check\_path: /admin/login\_check

failure\_path: null

logout:

path: /admin/logout

target: /admin/login

anonymous: true

*# -> end custom configuration*

*# default login area for standard users*

*# This firewall is used to handle the public login area*

*# This part is handled by the FOS User Bundle*

main:

pattern: .\*

context: user

form\_login:

provider: fos\_userbundle

login\_path: /login

use\_forward: false

check\_path: /login\_check

failure\_path: null

logout: true

anonymous: true

Add role hierarchy and provider, if you are not using ACL also add the encoder:  
*# config/packages/security.yaml*

security:

role\_hierarchy:

ROLE\_ADMIN: [ROLE\_USER, ROLE\_SONATA\_ADMIN]

ROLE\_SUPER\_ADMIN: [ROLE\_ADMIN, ROLE\_ALLOWED\_TO\_SWITCH]

SONATA:

- ROLE\_SONATA\_PAGE\_ADMIN\_PAGE\_EDIT *# if you are using acl then this line must be commented*

encoders:

FOS\UserBundle\Model\UserInterface: bcrypt

providers:

fos\_userbundle:

id: fos\_user.user\_provider.username

*# config/packages/security.yaml*

security:

access\_control:

*# Admin login page needs to be accessed without credential*

- { path: ^/admin/login$, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/admin/logout$, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/admin/login\_check$, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/admin/resetting, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

*# Secured part of the site*

*# This config requires being logged for the whole site and having the admin role for the admin part.*

*# Change these rules to adapt them to your needs*

- { path: ^/admin/, role: [ROLE\_ADMIN, ROLE\_SONATA\_ADMIN] }

- { path: ^/.\*, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

## **EXTENDING THE BUNDLE :**

bin/console sonata:easy-extends:generate SonataUserBundle --dest=src

cette commande va génere un nouveau bundle Application Application\Sonata\UserBundle\Entity\User.

Activer le bundle dans AppKernel

**new** Application\Sonata\UserBundle\ApplicationSonataUserBundle(),

Configurer fosuser et sonatauser pour recuperer le nouveau entite :

# config.yaml

fos\_user:

#...

user\_class: Application\Sonata\UserBundle\Entity\User

group:

group\_class: Application\Sonata\UserBundle\Entity\Group

#...

# config.yaml

sonata\_user:

class:

user: Application\Sonata\UserBundle\Entity\User

group: Application\Sonata\UserBundle\Entity\Group

enregister le bundle Application dans composer.json

php bin/console doctrine:schema:update –force

pour créer un utilisateur avec la cmd ligne pour fosuserbundle

bin/console fos:user:create adminuser -–super-admin