

## CREER UN USER EN SYMFONY .

Installer le bundle Symfony de sécurité : Symfony propose un bundle de sécurité qui simplifie la gestion des utilisateurs. Vous pouvez l'installer avec Composer en utilisant la commande suivante :

```
bash
```

```
composer require symfony/security-bundle
```

Créer une entité Utilisateur : Créez une entité utilisateur pour stocker les informations de l'utilisateur, telles que le nom d'utilisateur, le mot de passe, l'email, etc. Par exemple :

```
esyfony console make:entity User
```

Cela vous guidera pour créer l'entité.

Mettre à jour la base de données : Après avoir créé l'entité utilisateur, mettez à jour la base de données pour créer la table correspondante avec la commande Doctrine :

```
symfony console doctrine:migrations:diff
```

```
symfony console doctrine:migrations:migrate
```

Configurer le pare-feu de sécurité : Configurez le pare-feu de sécurité dans [security.yaml](#) pour spécifier les rôles, les stratégies d'authentification, etc. Exemple :

```
yaml
```

Copy code

```
security:
    # ...
    providers:
        app_user_provider:
            entity:
                class: App\Entity\User
                property: email
    firewalls:
        main:
            anonymous: ~
            form_login:
                login_path: app_login
                check_path: app_login
            logout:
                path: app_logout
                target: app_home
```

Créer des formulaires d'inscription et de connexion : Vous devrez créer des formulaires pour permettre aux utilisateurs de s'inscrire et de se connecter. Symfony fournit des commandes pour générer ces formulaires :

```
symfony console make:registration-form
```

```
symfony console make:login-form
```

Créer un contrôleur pour la gestion des utilisateurs : Créez un contrôleur pour gérer les actions liées aux utilisateurs, telles que l'inscription, la connexion, la modification de profil, etc.

Configurer la gestion des mots de passe : Symfony dispose d'un système de gestion des mots de passe intégré. Vous pouvez configurer comment les mots de passe sont encodés et vérifiés dans **security.yaml**.

Utiliser les annotations de sécurité : Utilisez des annotations de sécurité pour restreindre l'accès aux routes et aux contrôleurs en fonction des rôles des utilisateurs.

[Créer des templates Twig](#) : Créez des templates Twig pour afficher les pages d'inscription, de connexion et de profil.

### **POUR LE REGISTER**

Configurer le pare-feu de sécurité : Configurez le pare-feu de sécurité dans security.yaml pour spécifier les stratégies d'authentification, les fournisseurs de jetons, etc. Exemple :

yaml

Copy code

```
security:
    # ...
    firewalls:
        main:
            anonymous: ~
            guard:
                authenticators:
                    - App\Security\LoginFormAuthenticator
            logout:
                path: app_logout
                target: app_home
```

Créer un Authenticator : Vous pouvez utiliser l'authenticator que vous avez créé précédemment pour la connexion pour gérer également le processus d'inscription, ou vous pouvez créer un authenticator spécifique à l'inscription si nécessaire.

Configurer la route d'inscription : Assurez-vous d'avoir une route pour l'inscription configurée dans routes.yaml ou dans le fichier de routage de votre choix.

Créer un formulaire d'inscription : Vous devrez créer un formulaire d'inscription personnalisé. Vous pouvez le faire avec la commande :

```
symfony console make:form RegistrationFormType
```

Créer un contrôleur d'inscription : Créez un contrôleur pour gérer le processus d'inscription. Vous pouvez le faire avec la commande :

```
symfony console make:controller RegistrationController
```

Configurer la gestion des mots de passe : Symfony dispose d'un système de gestion des mots de passe intégré. Vous pouvez configurer comment les mots de passe sont encodés et vérifiés dans security.yaml.

Créer des templates Twig : Créez des templates Twig pour afficher la page d'inscription.

### **POUR LE LOGIN**

Configurer le pare-feu de sécurité : Configurez le pare-feu de sécurité dans security.yaml pour spécifier les stratégies d'authentification, les fournisseurs de jetons, etc. Exemple :

yaml

Copy code

```
security:
    # ...
    firewalls:
        main:
            anonymous: ~
            guard:
                authenticators:
                    - App\Security\LoginFormAuthenticator
            logout:
                path: app_logout
                target: app_home
```

**Créer un Authenticator** : Vous devez créer un authenticator pour gérer le processus d'authentification personnalisé. Vous pouvez le faire avec la commande :

```
symfony console make:auth
```

Cette commande générera un fichier [LoginFormAuthenticator.php](#) (ou un nom similaire) que vous devrez personnaliser selon vos besoins.

Configurer [les routes de connexion et de déconnexion](#) : Assurez-vous que vous avez des routes pour la connexion et la déconnexion configurées dans [routes.yaml](#) ou dans le fichier de routage de votre choix.

Créer un formulaire de connexion : Vous devrez créer un formulaire de connexion personnalisé si vous n'en avez pas déjà un. Vous pouvez le faire avec la commande :

```
symfony console make:form LoginFormType
```

Configurer la gestion des mots de passe : Symfony dispose d'un système de gestion des mots de passe intégré. Vous pouvez configurer comment les mots de passe sont encodés et vérifiés dans [security.yaml](#).

[Créer des templates Twig](#) : Créez des templates Twig pour afficher la page de connexion et de déconnexion.