

Documentation Technique

Authentification

Fichiers Modifiés

security.yaml

Ce fichier de configuration, situé dans le dossier config/packages, est crucial pour configurer la sécurité et l'authentification de votre application Symfony. Vous devez le modifier pour définir les fournisseurs d'utilisateurs, les firewalls, les règles d'accès, ainsi que les stratégies d'encodage des mots de passe. Il est également utile pour configurer de nouveaux processus ou méthodes d'authentification, déterminer où les utilisateurs sont chargés ou spécifier les restrictions d'accès.

Entité 'User'

La classe User, qui représente les utilisateurs de l'application, se trouve dans le dossier src/Entity. Elle implémente l'interface UserInterface de Symfony pour être compatible avec le système de sécurité de Symfony. Cette entité est essentielle pour la gestion des utilisateurs dans votre base de données. Il peut être nécessaire de la modifier pour ajouter des attributs à l'entité (par exemple, ajouter une adresse postale).

Controller SecurityController

La classe SecurityController, située dans le dossier src/Controller, gère les actions liées à la sécurité, telles que l'affichage du formulaire de connexion et le traitement de la déconnexion. Il peut être utile de la

modifier pour personnaliser le processus de connexion ou de déconnexion (par exemple, ajouter un captcha).

Fonctionnement de l'Authentification

L'authentification dans Symfony se déroule principalement à travers les firewalls définis dans le fichier de configuration `security.yaml`. Lorsqu'une requête est reçue, Symfony la compare aux patterns des firewalls pour déterminer si et comment l'authentifier.

Dans notre cas, nous utilisons un formulaire de connexion pour authentifier un utilisateur. Nous configurons le firewall en spécifiant le chemin du formulaire de connexion et le chemin de vérification. Symfony utilise les fournisseurs d'utilisateurs pour charger les informations de l'utilisateur pendant le processus d'authentification. Le résultat est un objet `User` qui est ensuite utilisé pour représenter l'utilisateur authentifié dans l'application.

Stockage des Utilisateurs

Les informations des utilisateurs sont stockées dans la base de données configurée via Doctrine ORM. Le mapping ORM de l'entité `User` détermine la table et les colonnes utilisées dans la base de données.