

# Documentation de l'Implémentation de l'Authentification avec Symfony et guide de contribution

## Introduction

Ce guide a pour objectif d'expliquer aux développeurs comment contribuer aux modifications du projet Symfony, en se concentrant sur l'implémentation de l'authentification. Vous découvrirez les fichiers à modifier, le processus d'authentification et le stockage des utilisateurs.

## Contribuer au Projet

Si vous souhaitez apporter des modifications au projet Symfony, suivez ces étapes :

### 1. Comprendre les Fichiers Clés

Avant de commencer, il est essentiel de comprendre quels fichiers sont impliqués dans l'authentification :

**SecurityController.php**: Ce fichier contient les actions liées à l'authentification, comme l'affichage du formulaire de connexion et la gestion des erreurs d'authentification. Vous pouvez le personnaliser pour modifier le comportement de l'authentification.

**security.yaml**: Ce fichier est au cœur de la configuration de sécurité de l'application. C'est ici que vous définissez les paramètres essentiels pour l'authentification et l'autorisation.

### 2. Modification des Fichiers

Une fois que vous avez identifié les fichiers à modifier, vous pouvez commencer à apporter des modifications :

Si vous souhaitez personnaliser le comportement de l'authentification, consultez

**SecurityController.php**.

Pour configurer les aspects essentiels de l'authentification, comme les hacheurs de mots de passe, les fournisseurs d'utilisateurs, les pare-feu et les autorisations, consultez

**security.yaml**.

### 3. Processus d'Authentification

Il est crucial de comprendre comment fonctionne le processus d'authentification :

L'utilisateur accède à la page de connexion de l'application.

Le formulaire de connexion est affiché par la méthode `loginAction` de `SecurityController.php`. Les erreurs d'authentification précédentes et le dernier nom d'utilisateur entré sont gérés ici.

L'utilisateur soumet le formulaire en entrant ses informations d'identification.

Symfony vérifie les informations d'identification soumises en interne en utilisant la configuration définie dans `security.yaml`. Si elles sont correctes, l'utilisateur est authentifié. Si les informations sont incorrectes, Symfony gère l'affichage des erreurs en utilisant la vue de connexion.

Une fois authentifié, l'utilisateur peut accéder aux pages restreintes en fonction des autorisations définies dans `security.yaml`.

## 4. Stockage des Utilisateurs

Les utilisateurs sont stockés dans la base de données de l'application, généralement dans une table `User`. La configuration de stockage des utilisateurs est spécifiée dans `security.yaml` sous la section `providers`.