

Authentification dans ToDo&Co

1. Fonctionnement

Le composant de sécurité de [Symfony](#) est utilisé, il contient de très riches fonctionnalités gérant authentications, autorisations, login/logout et le tout de façon normalisé. Il est configuré dans le fichier « security.yaml », on retrouve notamment :

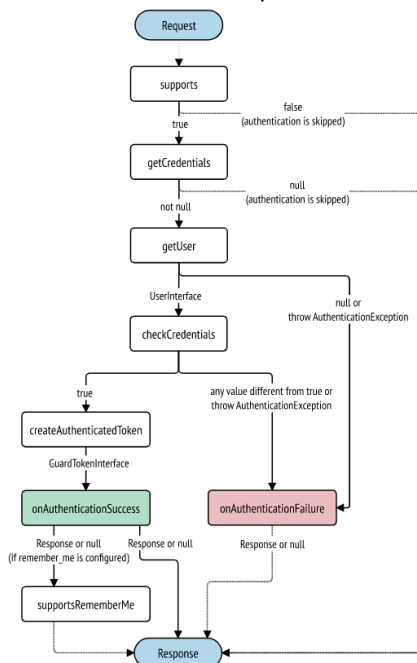
- Encodage du mot de passe.
- Provider (session, changement d'utilisateur).
- Firewall (authentification).
- Contrôle accès (rôle).

Le firewall est le système qui définit comment l'utilisateur va être authentifié. Il ne peut y avoir qu'un firewall en même temps pour traiter une requête. Mais la sécurité peut être différente entre le mode développement (dev) et le site en production (main).

2. Le firewall

Le fournisseur d'authentification utilisé s'appelle [Guard](#), il va être appelé à chaque requête et offre un contrôle complet de l'authentification.

Voici les méthodes présentes sur tous les authenticateurs et dans « LoginFormAuthenticator.php » :



- **supports(\$request)** = authentification nécessaire oui/non ?
- **getCredentials(\$request)** = lit et retourne le token.
- **getUser(\$credentials, \$userProvider)** = retourne l'utilisateur.
- **checkCredentials(\$credentials, \$user)** = vérifie si l'utilisateur est authentifié.
- **createAuthenticatedToken(\$user, \$providerKey)** = retourne le token.
- **onAuthenticationSuccess(\$request, \$token, \$providerKey)** = retourne un objet réponse après une authentification positive.
- **onAuthenticationFailure(\$request, \$exception)** = idem en cas d'authentification rejeté + raison dans l'exception.
- **start(\$request, \$authException)** = appelé en cas d'accès à une ressource nécessitant une authentification.
- **supportsRememberMe()** = to support "remember me" functionality.

https://symfony.com/doc/current/security/guard_authentication.html

3. Les utilisateurs

Les utilisateurs sont gérés par l'entité « Users » et sont stockés dans la base de données MySQL.

Le lien entre la classe User et la notion d'utilisateur de l'application est faite par la clé « User Provider » renseigné dans le fichier « Security.yaml ». C'est là qu'il est indiqué que l'identification se fait via le champ email.

```
# https://symfony.com/doc/current/security.html#where-do-users-come-from-user-providers
providers:
    # used to reload user from session & other features (e.g. switch_user)
    app_user_provider:
        entity:
            class: App\Entity\User
            property: email
```