Documentation Technique – Sécurité Web

Le projet intègre 6 types de sécurités, et le logger a été modifié pour l'environnement de prod :

Injection SQL :

D'origine, les injections SQL sont déjà maîtrisées dans Symfony, je n'ai pas utilisé de requête ou méthode personnalisée et ai gardé celle de base.

Utilisation de Slug :

Pour protéger la BDD, et les informations étant donné, toutes les URL affichent un slug à la place de l'ID des éléments. Cela est fait avec cette méthode :

```php
if($form->isSubmitted() && $form->isValid()){
    $slug = $slugger->slug($book->getTitle() . '-' . uniqid());
    $book->setSlug($slug);

    $em->persist($book);
    $em->flush();
}
```

En faisant cela, je n'affiche plus la clé primaire dans l'URL, cela peut empêcher de donner des informations qui pourraient permettre de faire un attaque ciblé sur un ou plusieurs éléments de la BDD.

Gestion des rôles par utilisateurs :

Les utilisateurs ne peuvent pas accéder au site tant qu'ils n'ont pas de compte :

```yaml
access_control:
    - { path: ^/register, roles: PUBLIC_ACCESS }
    - { path: ^/login, roles: PUBLIC_ACCESS }
    - { path: ^/book/edit, roles: ROLE_ADMIN }
    - { path: ^/book/delete, roles: ROLE_ADMIN }
    - { path: ^/, roles: ROLE_USER }
```

De plus seulement les admins peuvent créer, éditer ou supprimer les éléments du sites (ici des livres). Pour limiter des erreurs de redirections, des conditions ont été mise pour ne pas afficher les boutons si l'on est Admin :

```twig
{% if is_granted('ROLE_ADMIN') %}
    <a href="{{ path('book_edit', {'slug': book.slug}) }}">Modifier</a>

    <form action="{{ path('book_delete', {'id': book.id}) }}" method="post">
        <input type="hidden" name="csfr" value="{{csrf_token('delete' ~ book.id)}}">
        <input type="submit" value="Supprimer">
    </form>
{% endif %}
```

Limiter les inputs :

pour limiter les possibilités d'erreur involontaire, les forms ont été spécifié, et ont des règles strictes :

```
->add('Title', null, [
    'label' => 'Title',
    'required' => true,
    'attr' => [
        'placeholder' => 'Title of the Book'
    ]
])
->add('Price', NumberType::class, [
    'label' => 'Price',
    'required' => true,
    'attr' => [
        'placeholder' => 'Price of the Book'
    ],
    'html5' => true,
    'scale' => 2,
])
->add('Author', null, [
    'label' => 'Author',
    'required' => true,
    'attr' => [
        'placeholder' => 'Author of the Book'
    ]
])
->add('Pages', IntegerType::class, [
    'label' => 'Number of Pages',
    'required' => true,
    'attr' => [
        'placeholder' => 'Number of Pages'
    ]
])
```

En faisant cela, les inputs ne sont plus libres et nous pouvons d'ores et déjà contrôlé d'ores et déjà contrôlé les entré de données !

Token CSRF :

Utilisation d'un Token CSRF pour augmenter la sécurité :

```
<form action="{{ path('book_delete', {'id': book.id}) }}" method="post">
    <input type="hidden" name="csfr" value="{{csrf_token('delete' ~ book.id)}}">
    <input type="submit" value="Supprimer" onclick="return confirm('Êtes-vous sûr de vouloir supprimer ce Livre de la boutique ?')">
</form>
```

Le token permet de sécurisé le site et d'éviter que les requetes soit dévié, de plus j'ai ajouté une sécurité à la suppression ou l'utilisateur doit revalidé pour supprimé un élément.

Attaque XSS :

Pour limiter les attaques, j'ai placé au niveau des fonctions Set de la BDD des options supplémentaires :

```
$this->slug = htmlspecialchars($slug);
```

En plus de cela au moment de l'affichage des données en front, je contrôle cela :

```
<h2>{{book.title|raw}}</h2>
<p>{{book.author|raw}}</p>
<p>{{book.price|raw}}</p>
<p>{{book.pages|raw}}</p>
```

De plus j'ai quelque peux modifier le logger d'origine, pour l'environement de prod :

```yaml
rotated:
    type: rotating_file
    path: "%kernel.logs_dir%/%kernel.environment%.log"
    level: debug
    max_files: 10
```

Cela permet de d'écrire les logs sur des fichiers différents chaque jour, en faisant cela on peut éviter de perdre toutes les logs, et l'on a toujours une trace. Cela est réécris périodiquement sur 10 jours glissants.