Le framework Spring

Module 7 – Spring Security - Introduction

Objectifs

- La sécurité : généralités
- Où Spring gère la sécurité ?
- Les différents types de paramétrage
- Les mécanismes d'authentification

- Authentification
- Habilitation
- Intégrité
- Confidentialité
- Audit
- Non répudiation

Authentification :

→Les systèmes d'authentification et de gestion des sessions sont des fonctionnalités critiques des applications web.

→L'authentification

permet de s'assurer que seuls des utilisateurs légitimes peuvent accéder à l'application

→tandis que le mécanisme des sessions assure le suivi des diverses actions réalisées par les utilisateurs sur l'application.

Habilitation

→Définir des profils d'habilitation dans les systèmes (Rôles) en séparant les tâches et les domaines de responsabilité. Pour limiter l'accès des utilisateurs.

→Supprimer les permissions d'accès des utilisateurs dès qu'ils ne sont plus habilités à accéder à une ressource.

Intégrité

→C'est garantir que les données sont bien celles que l'on croit être ;

Confidentialité

→ Consiste à assurer que seules les personnes autorisées aient accès aux ressources échangées

Audit

→ Annuellement, il faut vérifier toutes les autorisations, habilitations et les données stockées.

→ Résilier si besoin les droits de certains utiliateurs.

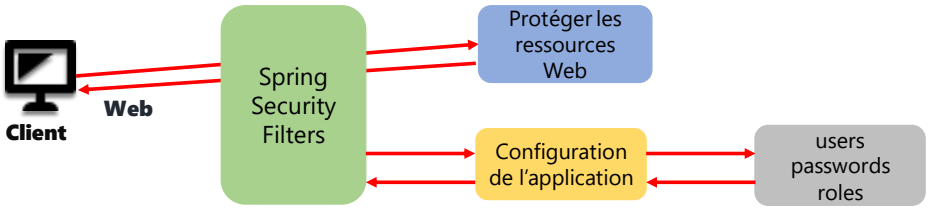
Non répudiation

→ Permet de garantir qu'une transaction ne peut être niée. Un utilisateur a modifié une donnée,

→il est possible de tracer tous ses comportements dans l'application.

Spring Security ?

- Spring Security couvre deux points de sécurité :
  - Authentification
  - Habilitation
- Les autres ne sont pas forcément du ressort de l'applicatif
- Un élément incontournable dans une plateforme de développement complète
  - Première version officielle Spring Security V2 sorti en avril 2008



Spring Security couvre deux points de sécurité :

Authentification

Habilitation

Les autres ne sont pas forcément du ressort de l'applicatif

Comme le modèle de sécurité standard de la plateforme Java EE.

→Mais un peu différemment…

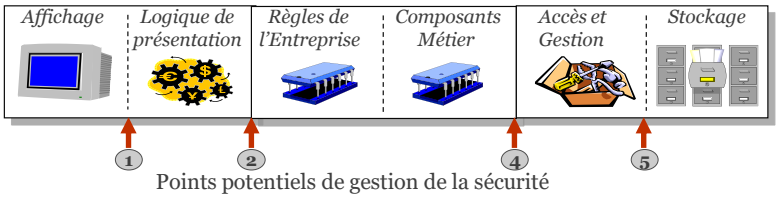
Un élément incontournable dans une plateforme de développement complète

→Spring Security est lancé fin 2003 sous le nom de «Acegi Security» par Ben Alex

→La première version officielle de Spring est Spring Security V2 sorti en avril 2008

Où gérer les habilitations ?

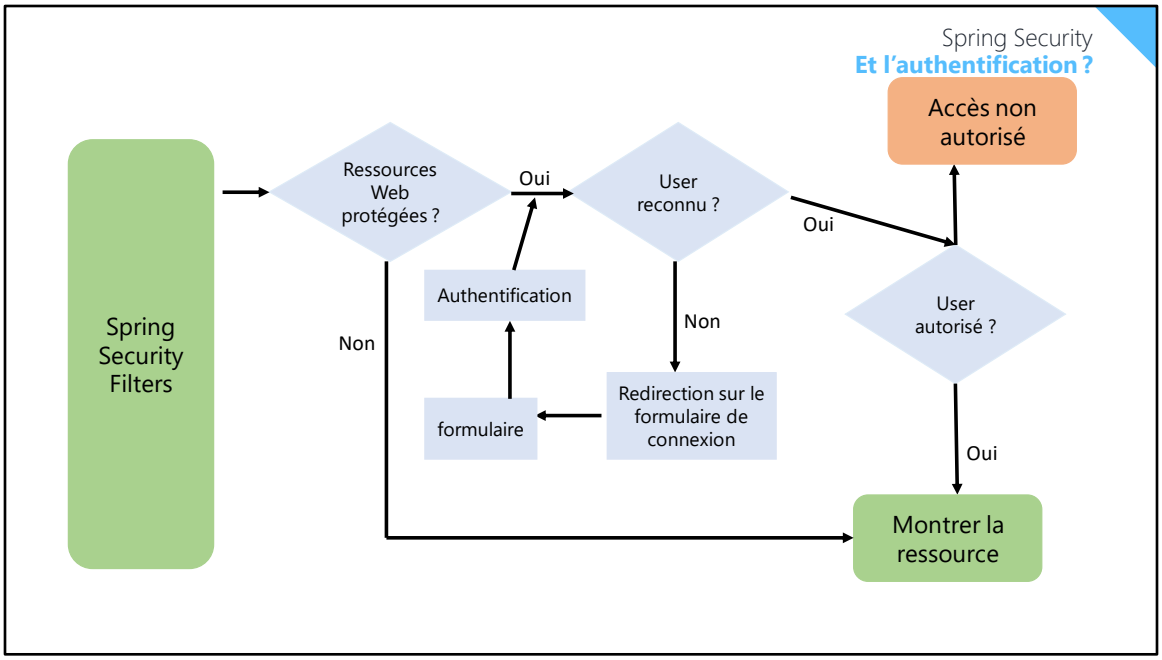
- Les habilitations peuvent se gérer à divers endroits :



Points potentiels de gestion de la sécurité

- La gestion des habilitations dans les couches hautes n'empêche pas la gestion dans les couches basses : il est nécessaire de véhiculer un contexte de sécurité entre les couches !

Il est possible de sécuriser dès la couche présentation.
Il est nécessaire d'appliquer les règles d'habilitations dans les couches basses pour protéger l'architecture complète.
La sécurité peut être mise en place sur une API REST, il y aura un token pour l'utilisateur et les habilitations sur Spring Data REST.



Configuration des contraintes de sécurité par classe @Configuration

Spring Security fournit pour les application Spring MVC, un formulaire par défaut de connexion.

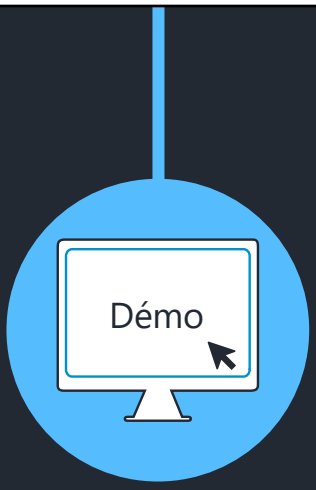
Il est possible de déclarer l'authentification et les autorisations par :

- Le code directement (pour les tests)
- JDBC
- LDAP
- Solution spécifique, ….

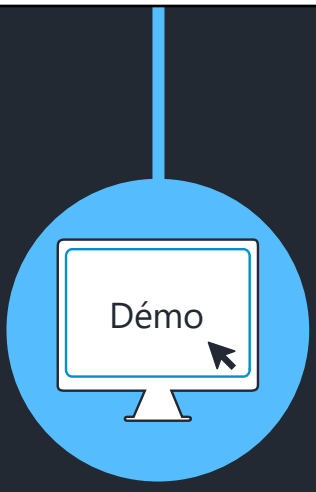
- Ajout des starters :

```
dependencies {
//Starter Spring Security
implementation 'org.springframework.boot:spring-boot-starter-security'
```

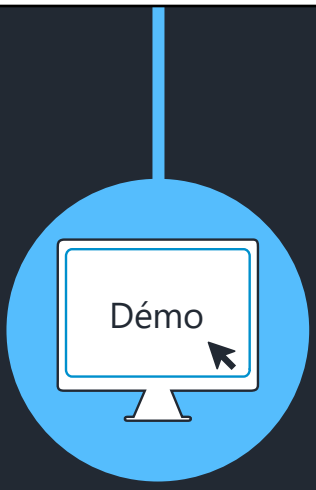
- Création d'une classe de configuration
  - Permet de préciser la stratégie pour les utilisateurs
  - Gestion des URLs et des permissions dessus



Configuration Spring Security – Utilisateur dans le code



Configuration Spring Security avec une Base de données



Spring Boot Security et Spring Boot REST API et authentification de base