Security

Petr Křemen and Lama Saeeda

petr.kremen@fel.cvut.cz, saeeda.lama@fel.cvut.cz

Winter Term 2022

Contents

Spring Security

Spring Security

Spring Security offers the following annotations:

- `@PreFilter` for filtering input iterable argument based on security constraints expressed in SpEL.
- `@PostFilter` for filtering output iterable value based on security constraints expressed in SpEL.
- `@PreAuthorize` for authorizing method execution based on security constraints expressed in SpEL.
- `@PostAuthorize` for authorizing return from the method execution based on security constraints expressed in SpEL.

Spring Security - SpEL

Relevant SpEL expressions:

- `hasRole('ROLE_ADMIN')` checks whether the currently logged in user has the 'ADMIN' role.
- `and`, `or` logical operators.
- `principal` the currently logged in user, e.g., `principal.username`.
- `filterObject` the object filtered from the collection, e.g., `filterObject.customer`.

Become familiar with these annotations before starting the following tasks.

Security in E-shop

Important security-related classes and notions used in the e-shop involve

- `SecurityConfig` and all the beans used in it
- `SecurityUtils`
  - Namely, the thread local-based `SecurityContextHolder`
- Aforementioned Spring security annotations

Tasks

Syncing Your Fork

1. Ensure you have no uncommitted changes in the working tree
   - `git status` → your branch is up to date, nothing to commit
2. Fetch branches and commits from the upstream repository (EAR/B221-eshop)
   - `git fetch upstream`
3. Check out the task branch from **upstream** (one line!)
   - `git checkout -b b221-seminar-10-task upstream/b221-seminar-10-task`
4. Do the task
5. Commit and push the solution to **your** fork
   - `git push -u origin b221-seminar-10-task`

Task – 1 point

- Implement `DefaultAuthenticationProvider.authenticate` method to return the expected authentication information.
- **Acceptance criteria:** Tests in `DefaultAuthenticationProviderTest` should pass.
- **Hints:** Use `SecurityUtils`, `UserDetails`, and `UserDetailsService` classes in the method implementation.

Task – 1 point

1. Data modifying operations in `CategoryController` should be allowed only for administrators. Use appropriate Spring security features to secure the corresponding endpoints.
2. Method `OrderService.findAll` should return only the orders of the current user or all orders if the current user is administrator.

Acceptance criteria:

1. All tests in `CategoryControllerSecurityTest` pass.
2. All tests in `OrderServiceSecurityTest` pass.

Hints: Administrators are users with `Role.ADMIN`.

Resources

- https://docs.spring.io/spring/docs/current/spring-framework-reference/core.html#expressions
- https://docs.spring.io/spring-security/site/docs/current/reference/html5/#el-access