Reflektion över individuell laboration i Java Web Services

Fredrik Planck

Syfte

Den här laborationen kretsade kring säkerhet, där uppgiften var att bygga ett säkert ”blogg-REST API” med Spring Boot . Användare kan hämta eller lägga till bloggposter och info om dessa.

Säkerhet

För autentisiering användes KeyCloak som kördes i en Docker-container. Vid inloggning via Postman med giltiga credentials erhöll användaren ett Json Web Token (JWT) från KeyCloak, giltigt i 300 sekunder och som förlängs varje gång det används. Tokenet fungerar som en biljett som ger användaren åtkomst till API:et. API:ets auktorisering hanteras av Spring Security, som med hjälp av användarnas tilldelade roller i JWT bestämmer vilka endpoints respektive användare ska ha tillgång till.
Säkerhetsinställningarna hanteras i en separat SecurityConfig-klass där de olika endpointsens skydd finns definierat med hjälp av SecurityFilterChain.

Design

Jag använde mig av MVC-strukturen, och återanvände delar av kod från ett tidigare REST API-projekt. Jag försökte också lägga till lite nya saker, t.ex en GlobalAccessHandler för att hantera hur felmeddelanden och liknande presenteras för användaren. Vid uppdatering av blogginlägg användes en DTO-klass, för att undvika att användaren skulle kunna uppdatera annat än ”title” och ”content”.

Reflektioner

Det var mycket intressant att se hur JWT-tokens skapas och används ”bakom kulisserna”. Jag har i många år använt diverse tokens som slutanvändare, men aldrig haft någon insikt i hur det faktiskt går till när de skapas. Den största utmaningen var att få till JwtAuthConverter att fungera ihop med KeyCloak, vilket löste sig med hjälp av den förinspelade videon.
I övrigt uppstod förvirring när det inte gick att skicka egna felmeddelanden, utan det enda som visades för användaren var ”403 Forbidden”. Efter en del googlande var lösningen att lägga till ”.requestMatchers(”/error”).permitAll()” i SecurityFilterChain. Ett annat problem jag kämpade jag mycket och länge med deletepost-endpointen och förstod inte varför jag hela tiden fick ”403 Forbidden” tills jag upptäckte… att jag hade angett ”v3” istället för ”v2” i URL:en.
Slutligen tycker jag att laborationen har lärt mig många nya saker och jag har fått en bättre förståelse för autentisering och auktorisering i praktiken.