

Spring Security

SpringBoot 기준으로 서술

[Java](#), [Spring](#)

Overview

- Authentication : 인증. “유효한 사용자인가?”
- Authorization : 허가(권한 부여). 유효한 사용자가 이 작업을 할 수 있는가?



Documents

- [Spring Security Architecture](#)
- [\[Docs\] Spring Security - Servlet Applications](#)
- [Samples](#)

Authentication

Docs

- ProviderManager는 Class, 이외는 모두 Interface

classDiagram class AuthenticationManager { authenticate(Authentication authentication) Authentication } class AuthenticationProvider { authenticate(Authentication authentication) Authentication supports(Class~?~ authentication) boolean } class UserDetailsService { loadUserByUsername(String username) UserDetails } class UserDetails { getAuthorities() Collection~? extends GrantedAuthority~ getPassword() String isAccountNonExpired() boolean isAccountNonLocked() boolean isCredentialsNonExpired() boolean isEnabled() boolean } AuthenticationProvider <|-- ProviderManager AuthenticationManager --> AuthenticationProvider : authRequest AuthenticationManager <.. UserDetailsService : 서비스 주입 후 authenticate()에서 사용자 정보를 꺼내어 사용 UserDetailsService -- UserDetails

- [AuthenctaionManager](#)
 - [AuthenticationManagerBuilder](#) : [WebSecurityConfigurerAdapter](#) 의 [configure\(AuthenticationManagerBuilder auth\)](#) Method로 “auth” 인자를 사용할 수 있는데, 이 인자의 “userDetailsService(T userDetailsService)”로 “UserDetailsService” 주입이 가능함.
- [AuthenticationProvider](#)
- [ProviderManager](#)
- [UserDetailsService](#)
 - [UserDetails](#)

Authorization

[Docs](#)

Protection Against Exploits

[Docs](#)

From:

<http://wiki.ledyx.xyz/> - **Ledyx WIKI**

Permanent link:

<http://wiki.ledyx.xyz/doku.php?id=back-end:spring:security&rev=1619533841>

Last update: **2021/04/27 14:30**

