技術筆記

筆記更新紀錄

|  |  |  |
| --- | --- | --- |
| 日期 yyyy/MM/dd | 更新人員 | 更新備註 |
| 2020/02/15 | Miles | 建立文件 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

目錄

[1關於OAuth 4](#_Toc32574543)

[2關於OAuth2 4](#_Toc32574544)

[2關於OAuth2 的版本 4](#_Toc32574545)

## 1關於Spring Security

|  |
| --- |
| Spring Security 是什麼? |
| Spring Security是一個基於Java / Java EE的框架，為企業應用程式提供身份驗證，授權和其他安全功能。為Spring的其中一個模組。 |

## 2剖析Spring Security 概念

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

## 2使用Spring Security(身分驗證管理器)

|  |
| --- |
| 1 導入Lib |
| 因為使用微服務，所以可以直接在Gradle導入    範例暫時使用2.1.3.RELEASE 版本。  如果非使用微服務，可導入以下Lib |

|  |
| --- |
| 2. 建立SpringSecurity Config類別 |
| 作為一個設定類別，要加上@Configuration  啟動SpringSecurity類別要加上@EnableWebSecurity  繼承WebSecutiryConfigurerAdapter 橋接器 |

|  |
| --- |
| 2.1 建立SpringSecurity Config類別 => 配置身分驗證管理器 |
| 當我們繼承了了WebSecutiryConfigurerAdapter 可以Overrid    配置我們要如何管理身分的驗證。  我們要配置存放這些身分的資訊在記憶體當中(可以存放在不同地方，之後會再研究)    配置編碼方式，Spring Secutiry5之後強制要求密碼需要進行加密，所以最基本的使用方式就是使用PasswrodEncoder 並且使用Spring 官方推薦的BCryptPasswordEncoder 進行加密。  /\*\* 配置密碼的編碼方式 spring secutiry5 之後強制要求密碼需要進行編碼 我們使用Spring 官方推薦的加密演算法方法BCryptPasswordEncoder \*/  PasswordEncoder pwdEncoder = new BCryptPasswordEncoder();  inMemoryUserDetailsManagerConfigurer.passwordEncoder(pwdEncoder);  我們叫建立使用者身分，該範例暫時把身份寫在程式裡。後續會根據不同情況取得身分。 |

|  |
| --- |
| 2.2 整個身分驗證配置大致上是這樣的… |
|  |

|  |
| --- |
| 2.3 配置Http攔截處理(基本做法後續會在研究) |
| 暫時先用以上的程式碼，該部分後續會再討論 |

|  |
| --- |
| 2.4 測試看看吧 |
| 當我們輸入了網址URL(再次聲明，我們使用SpringBoot並且建立了一個RESTController)  會出現Spring 官方提供的登入畫面。    登入後進會成功進入我們的RESTController的RequsetMapping!!! |

## 3使用Spring Security(身分驗證管理器-自定義使用者權限)

|  |
| --- |
| 1 建立SpringSecurity Config類別 => 配置身分驗證管理器(自定義使用者權限) |
| 當我們繼承了了WebSecutiryConfigurerAdapter 可以Override    配置我們要如何管理身分的驗證。  我們要配置存放這些身分的資訊在UserDetail當中，所以可以使用UserDetailsService()，並且該Service會回傳UserDetails這個類別。    當我們建立UserDetails的時候要傳入使用者的名稱(username)並且傳入密碼與權限範圍。  而權限範圍可以是一個Array。  我們也可以自定義使用者權限 |