

Ejemplo de autenticación con Google

Registra tu aplicación en Google:

- Obtén el **client ID** y **client secret** de cada proveedor registrando tu aplicación en:
 - [Google Developers Console](#)
 - [Facebook for Developers](#)

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-oauth2-client</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-security</artifactId>
</dependency>
```

Configura los proveedores en **application.yml** o **application.properties**

```
spring:
  security:
    oauth2:
      client:
        registration:
          google:
            client-id: TU_GOOGLE_CLIENT_ID
            client-secret: TU_GOOGLE_CLIENT_SECRET
            scope:
              - email
              - profile
          facebook:
            client-id: TU_FACEBOOK_CLIENT_ID
            client-secret: TU_FACEBOOK_CLIENT_SECRET
            scope:
              - email
              - public_profile
```

Puedes ofrecer enlaces directos para iniciar sesión con Google o Facebook:

```
<a href="/oauth2/authorization/google">Login con Google</a>
<a href="/oauth2/authorization/facebook">Login con Facebook</a>
```

En un controlador puedes obtener los datos del usuario autenticado así:

```

import org.springframework.security.core.annotation.AuthenticationPrincipal;
import org.springframework.security.oauth2.core.user.OAuth2User;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class SocialLoginController {

    @GetMapping("/user")
    public String getUserInfo(@AuthenticationPrincipal OAuth2User principal) {
        return "Hola, " + principal.getAttribute("name");
    }
}

```

Configura reglas de seguridad personalizadas (opcional)

```

@Configuration
@EnableWebSecurity
public class SecurityConfig extends WebSecurityConfigurerAdapter {
    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http
            .authorizeRequests()
                .antMatchers("/", "/login").permitAll()
                .anyRequest().authenticated()
            .and()
                .oauth2Login()
                    .loginPage("/login");
    }
}

```