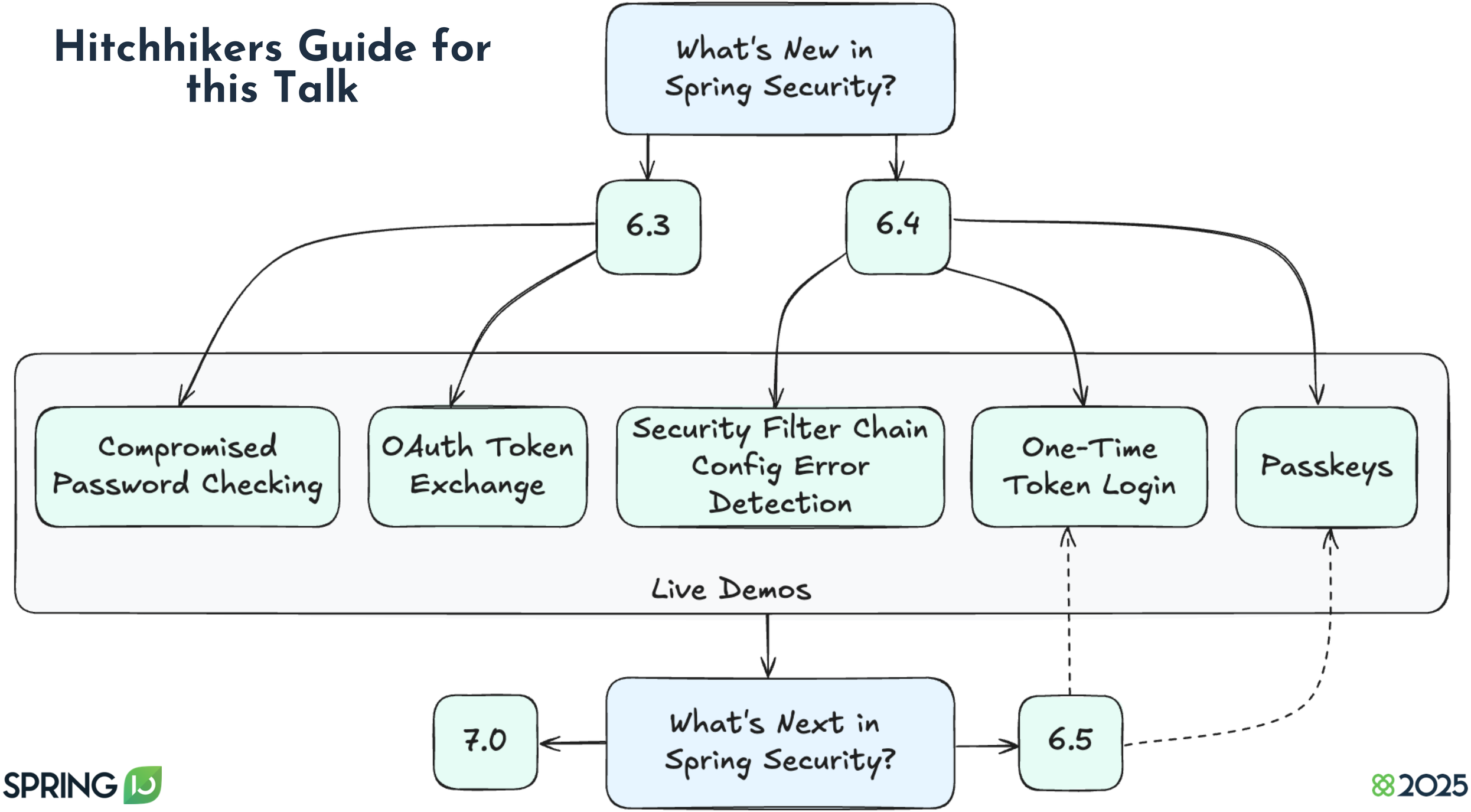
MODERN AUTHENTICATION DEMYSTIFIED:
A Deep Dive into Spring Security's Latest Innovations

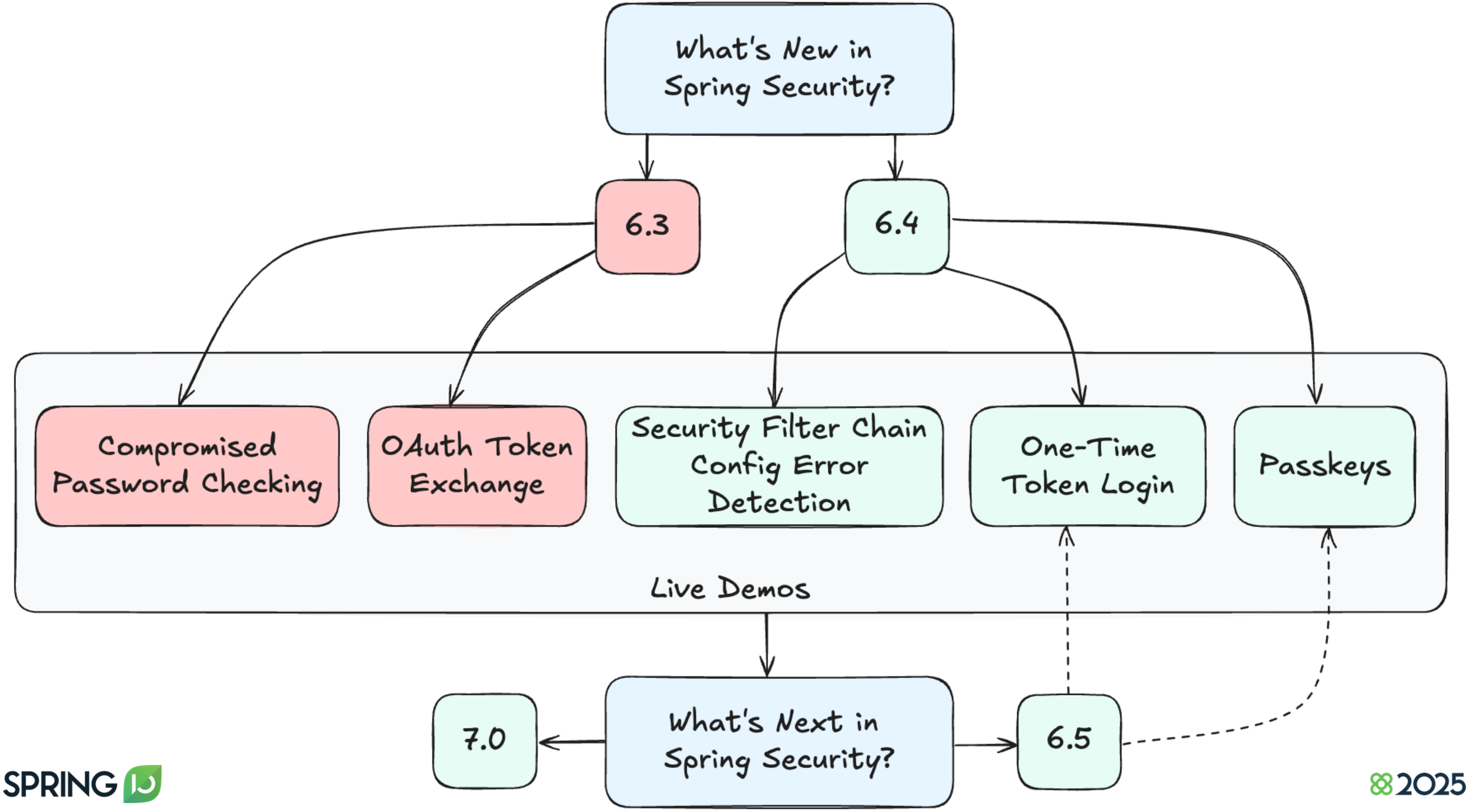
Andreas Falk
@andifalk

About Me

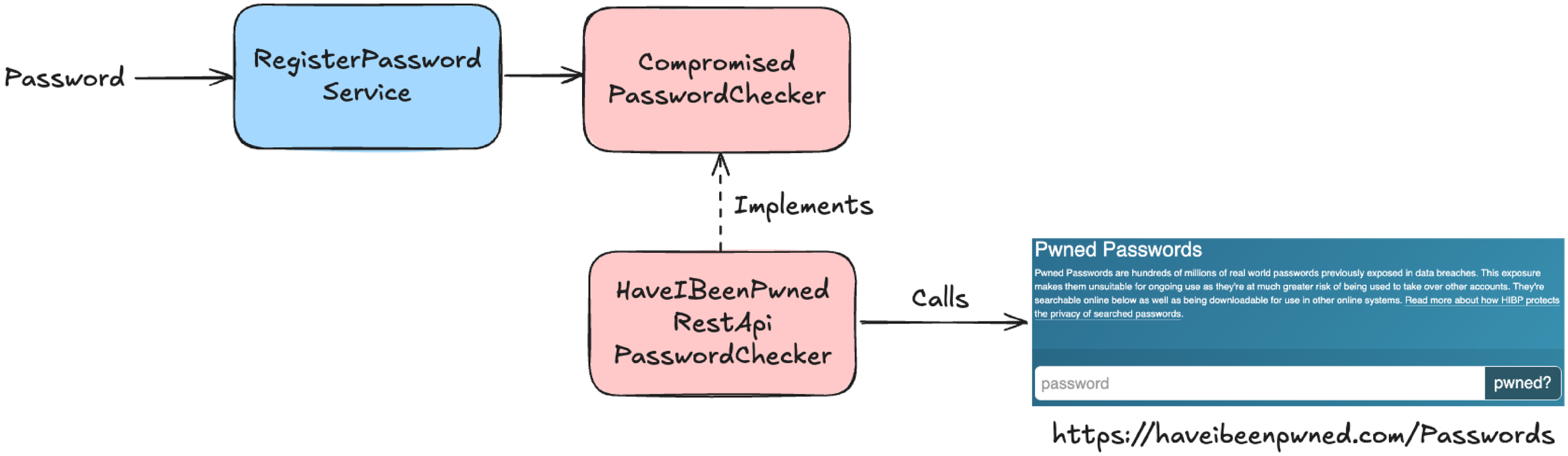
https://www.linkedin.com/in/andifalk

Hitchhikers Guide for this Talk




What's New in Spring Security?
6.3
6.4
Compromised Password Checking
OAuth Token Exchange
Security Filter Chain Config Error Detection
One-Time Token Login
Passkeys
Live Demos
What's Next in Spring Security?
7.0
6.5

Compromised Password Checking (Spring Security 6.3)



https://docs.spring.io/spring-security/reference/6.3/features/authentication/password-storage.html#authentication-compromised-password-check

DEMO

https://github.com/andifalk/whats-new-in-spring-security

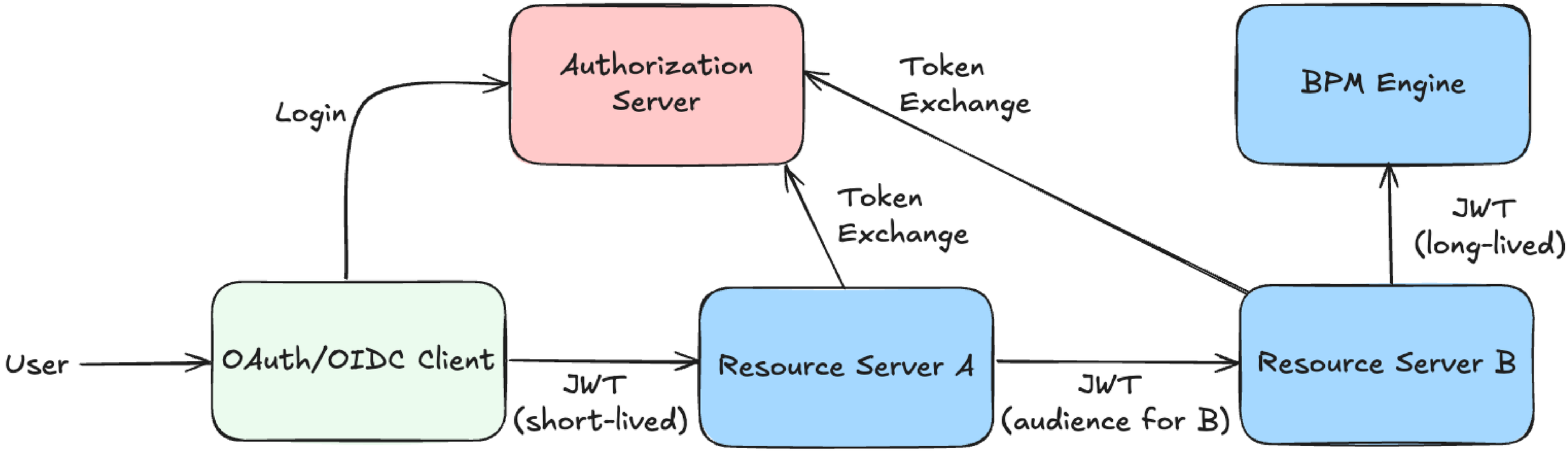
Compromised Password Checking (Spring Security 6.3)

Even Better…

Get Rid Of
Passwords completely 😉

→ See OAuth 2.1 just next & Passkeys a bit later…

OAuth Token Exchange (Spring Security 6.3)



https://docs.spring.io/spring-security/reference/6.3/whats-new.html#_oauth_2_0_token_exchange_grant_5199

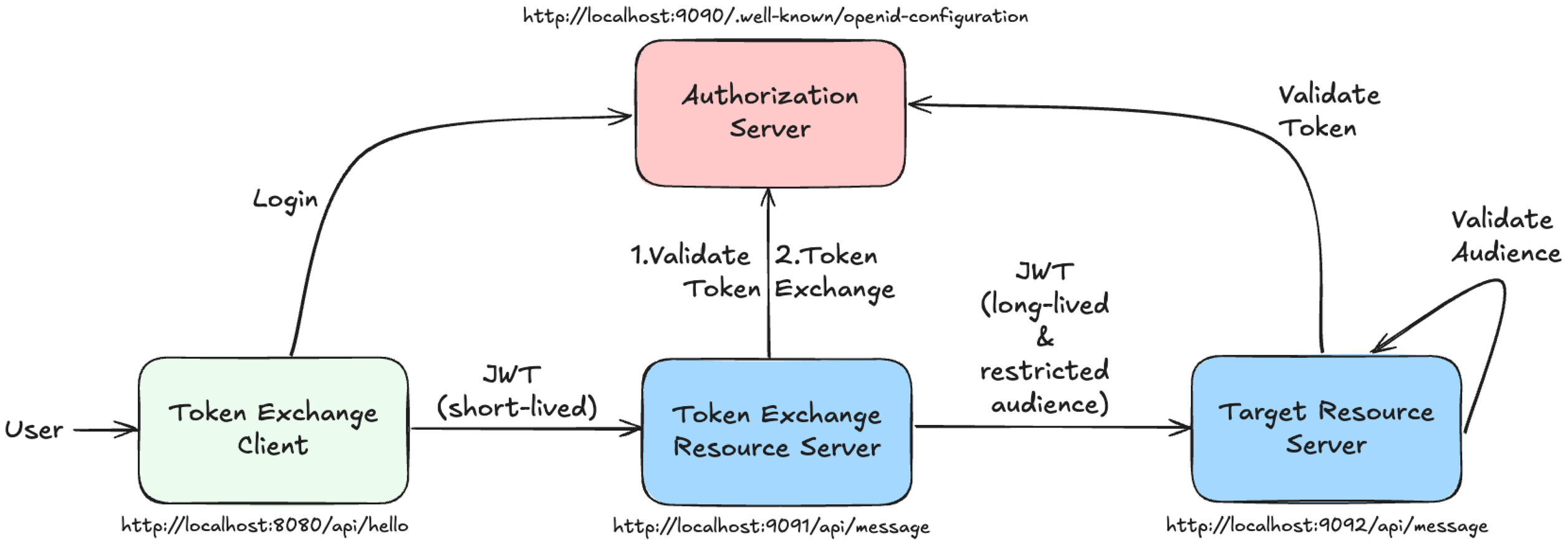
Impersonation vs. Delegation in OAuth Token Exchange

⚖️ Comparison Table (OAuth Token Exchange)

Feature	Impersonation	Delegation
RFC 8693 Usage	Only `subject_token`	`subject_token` + `actor_token`
Identity in access token	Subject only (`sub=user`)	Subject + Actor (`sub=user`, `act=caller`)
Use case example	Login as user / Support admin	Microservice acting on behalf of a user
Auditability	Limited – appears as user only	Full – includes caller identity
Security risk	Higher – hides real caller	Lower – maintains trust chain

https://www.rfc-editor.org/rfc/rfc8693.html

OAuth Token Exchange - Demo Scenario



https://docs.spring.io/spring-security/reference/6.3/whats-new.html#_oauth_2_0_token_exchange_grant_5199

https://github.com/andifalk/whats-new-in-spring-security

Further Improvements in Spring Security 6.3

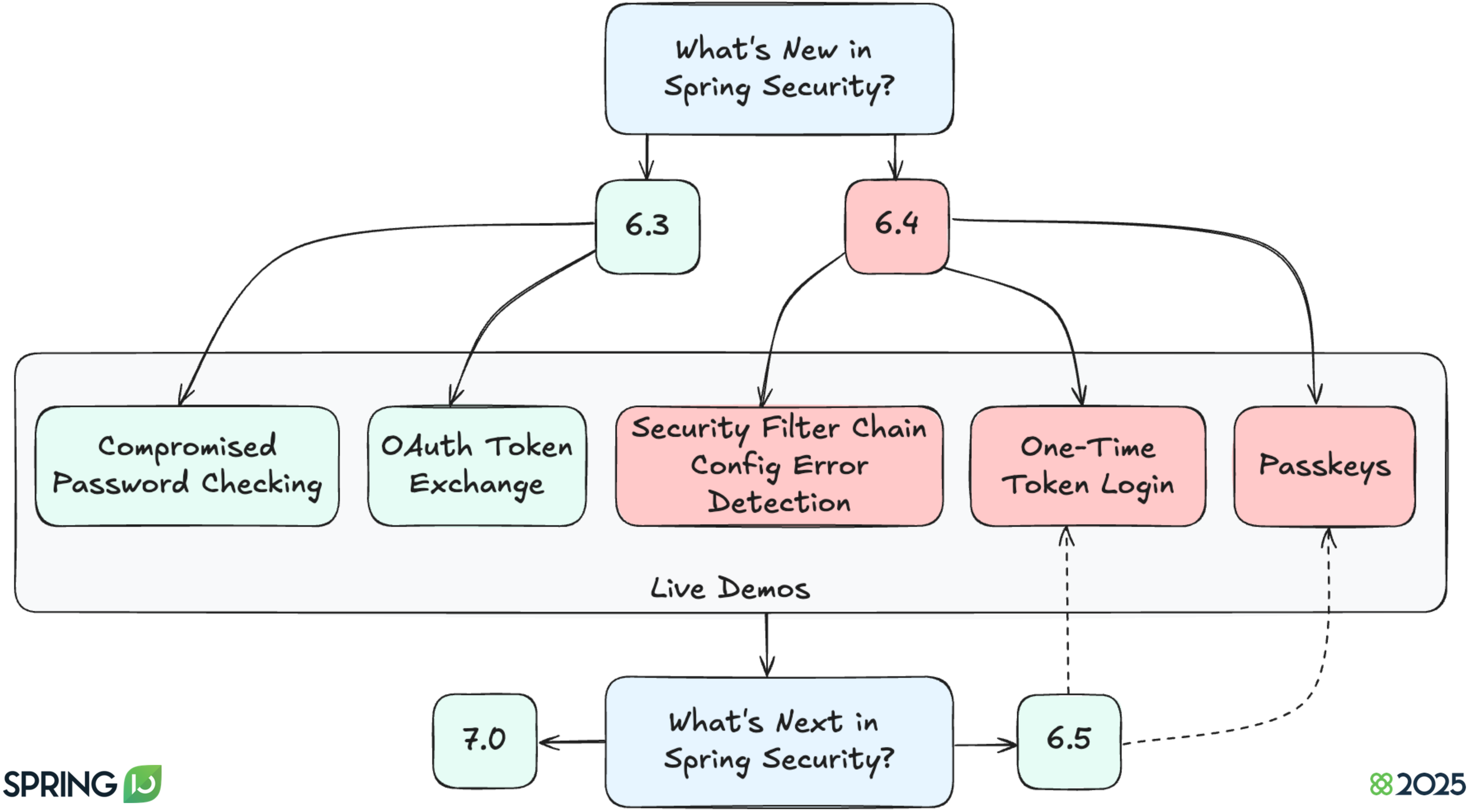
- Authorization
  - Annotation Parameters
  - Secure Return Values
  - Error Handling

```
@Retention(RetentionPolicy.RUNTIME)
@Target(ElementType.METHOD)
@PreAuthorize("hasRole('{role}')")
public @interface PreGetBankAccounts {
    String role();
}
```

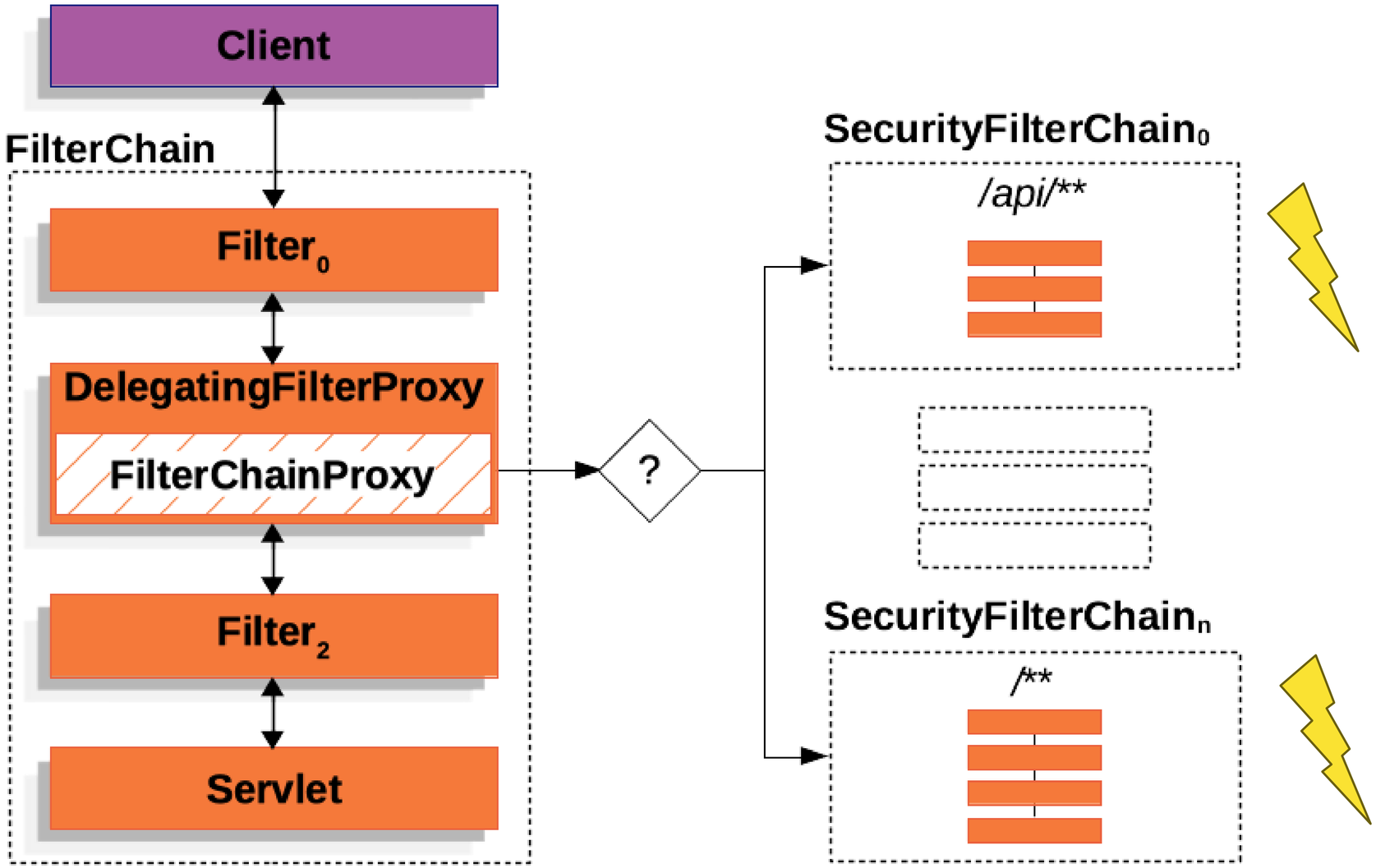
```
@Retention(RetentionPolicy.RUNTIME)
@Target(ElementType.METHOD)
@AuthorizeReturnObject
public @interface PostReadBankAccount {}
```

```
@Component
public class MaskMethodAuthorizationDeniedHandler implements MethodAuthorizationDeniedHandler {
    @Override
    public Object handleDeniedInvocation(
        MethodInvocation methodInvocation, AuthorizationResult authorizationResult) {
            return "******";
    }
}
```

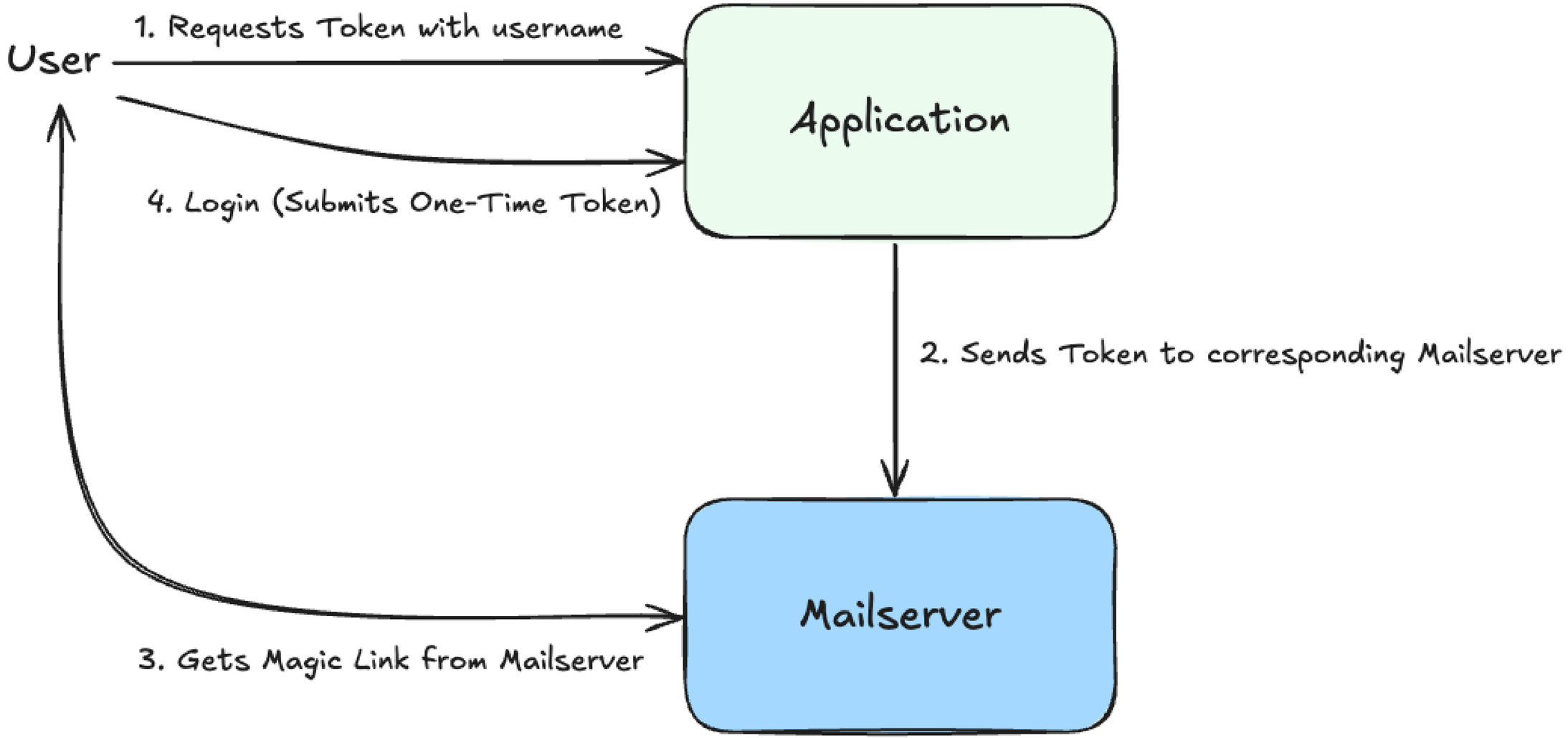
https://docs.spring.io/spring-security/reference/6.3/whats-new.html


What's New in Spring Security?
6.3
6.4
Compromised Password Checking
OAuth Token Exchange
Security Filter Chain Config Error Detection
One-Time Token Login
Passkeys
Live Demos
7.0
What's Next in Spring Security?
6.5

Security Filter Chain Error Detection (Spring Security 6.4)



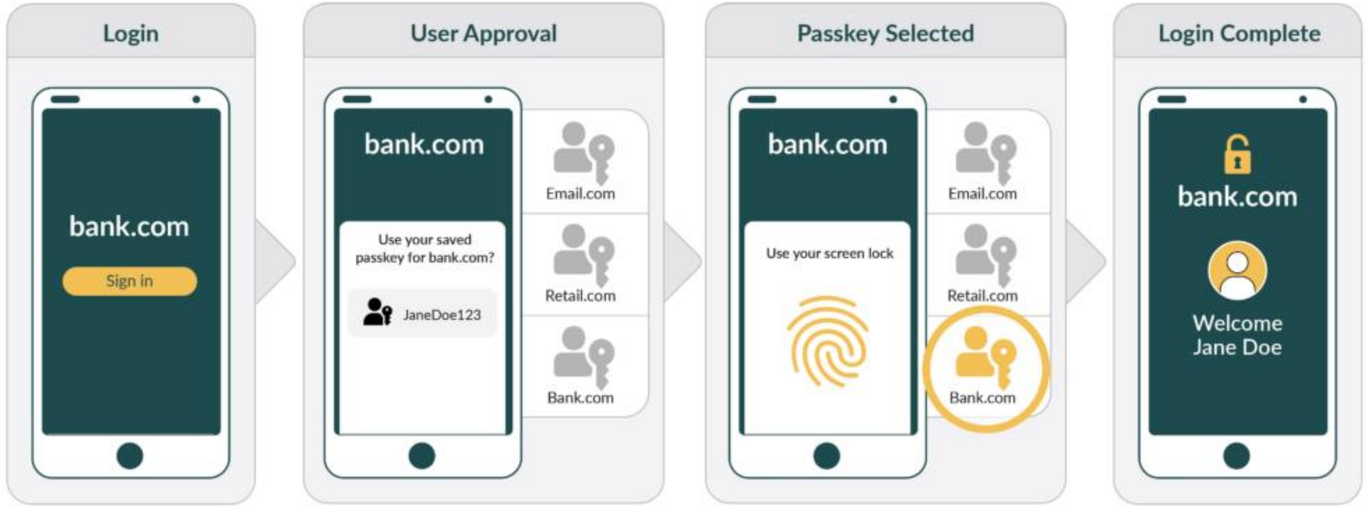
One-Time Token Login (Spring Security 6.4)

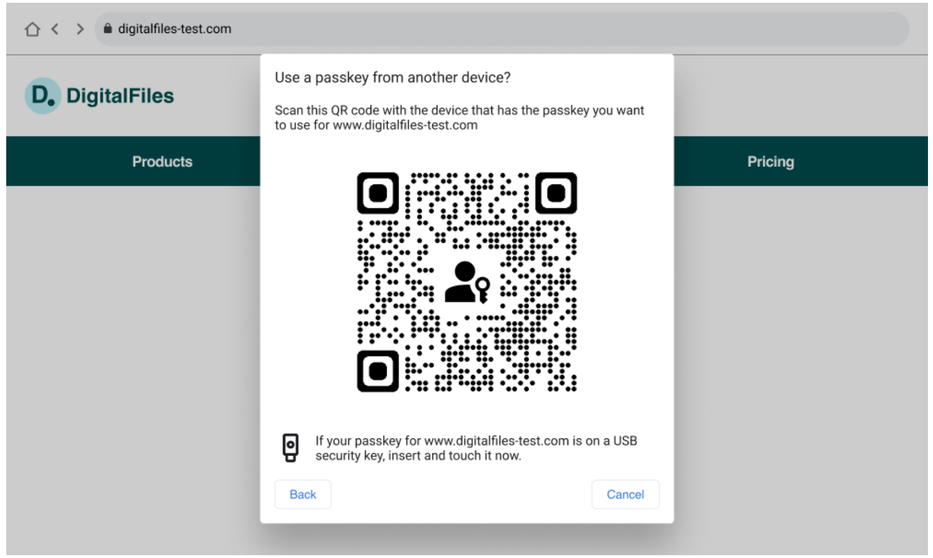


https://docs.spring.io/spring-security/reference/whats-new.html#_one_time_token_login

Passkeys (Spring Security 6.4)

- Provides faster, easier, and more secure sign-ins to websites and apps
- Works across user's devices.
- Strong and phishing-resistant.
- Unique cryptographic public/private key pairs (passkeys) to every online service
- Replacement for Passwords





https://docs.spring.io/spring-security/reference/servlet/authentication/passkeys.html
https://webauthn4j.github.io/webauthn4j/en/

Passkeys Support and Relation to WebAuthn & FIDO2

WebAuthn
The **Engine**

Passkeys
The **user-friendly car built around it**

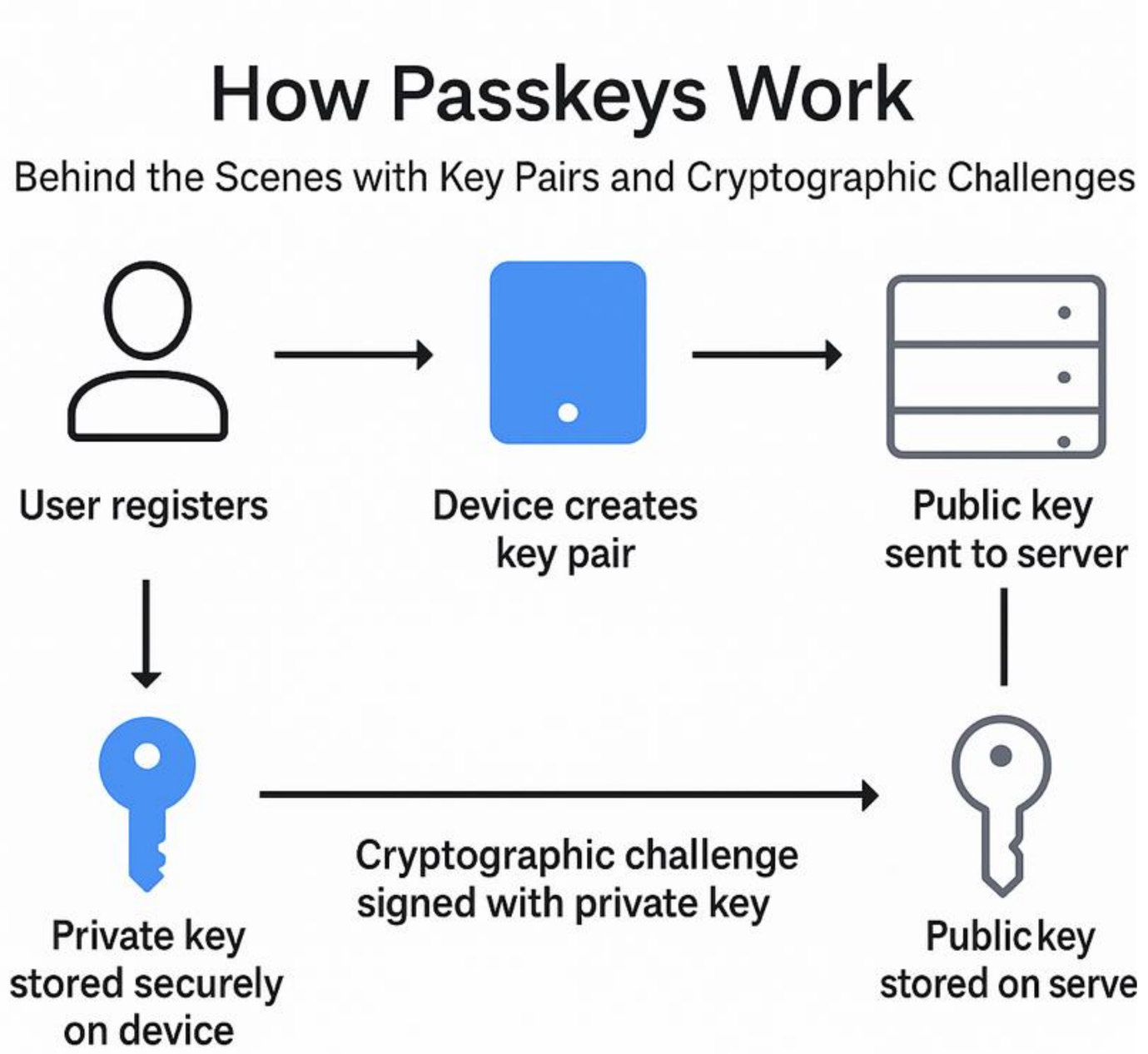
Passkeys are **FIDO2** sign-in credentials

How Passkeys Work on Different Operating Systems

OS / Device	Where Passkeys Are Stored	Sync Across Devices?
macOS	iCloud Keychain	✅ Yes (via iCloud)
iOS	iCloud Keychain	✅ Yes
Windows 11	Windows Hello / Credential Manager	⚠️ No (yet)
Android	Google Password Manager	✅ Yes (via Google Account)
Linux	Usually requires external authenticator (e.g. YubiKey)	❌ No built-in sync
Chrome OS	Google Account (like Android)	✅ Yes

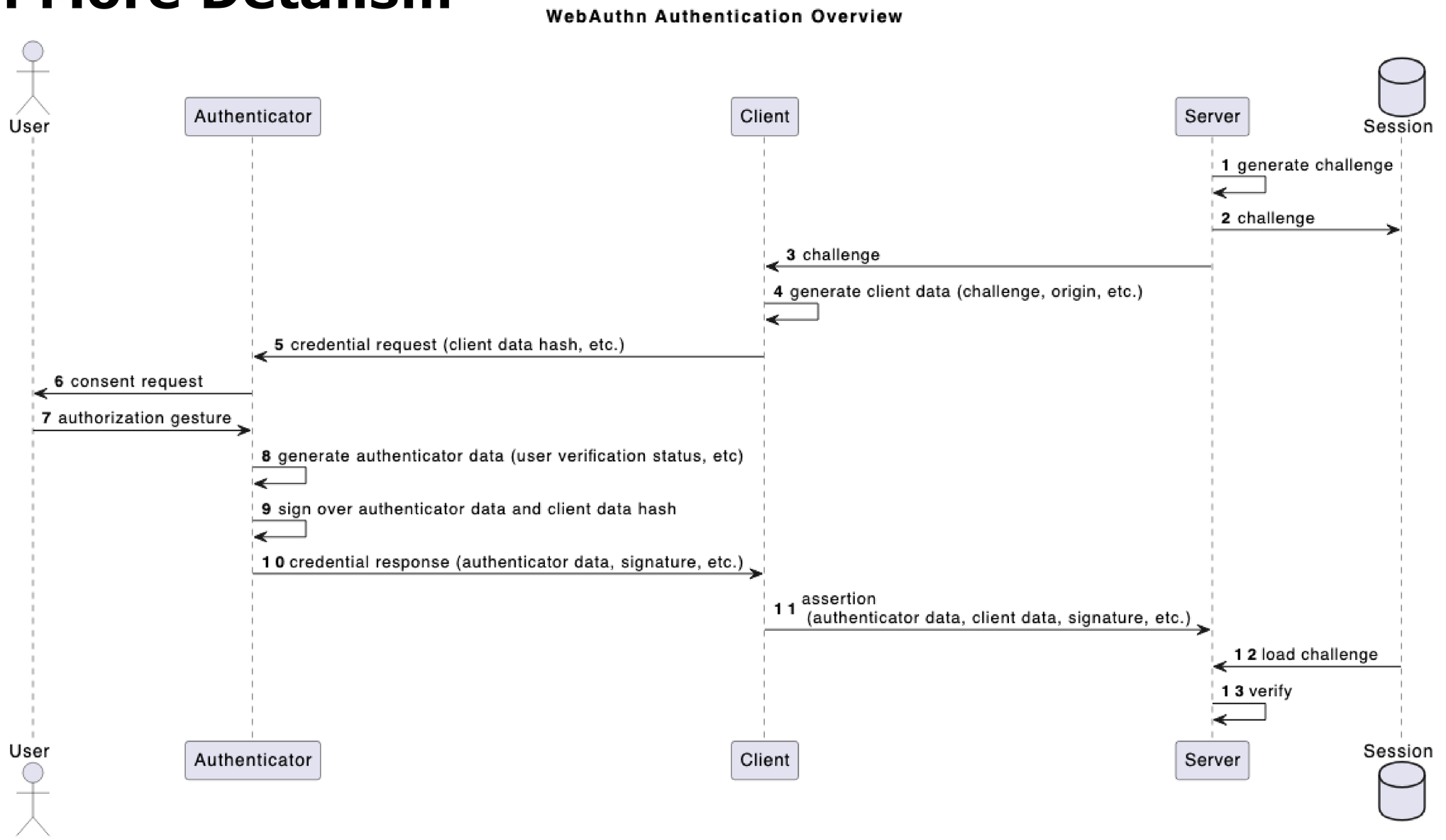
1.Register

2.Login



https://www.w3.org/TR/webauthn-2/
https://fidoalliance.org/passkeys

Even More Details...



https://webauthn4j.github.io/webauthn4j/en/

Further Improvements in Spring Security 6.4

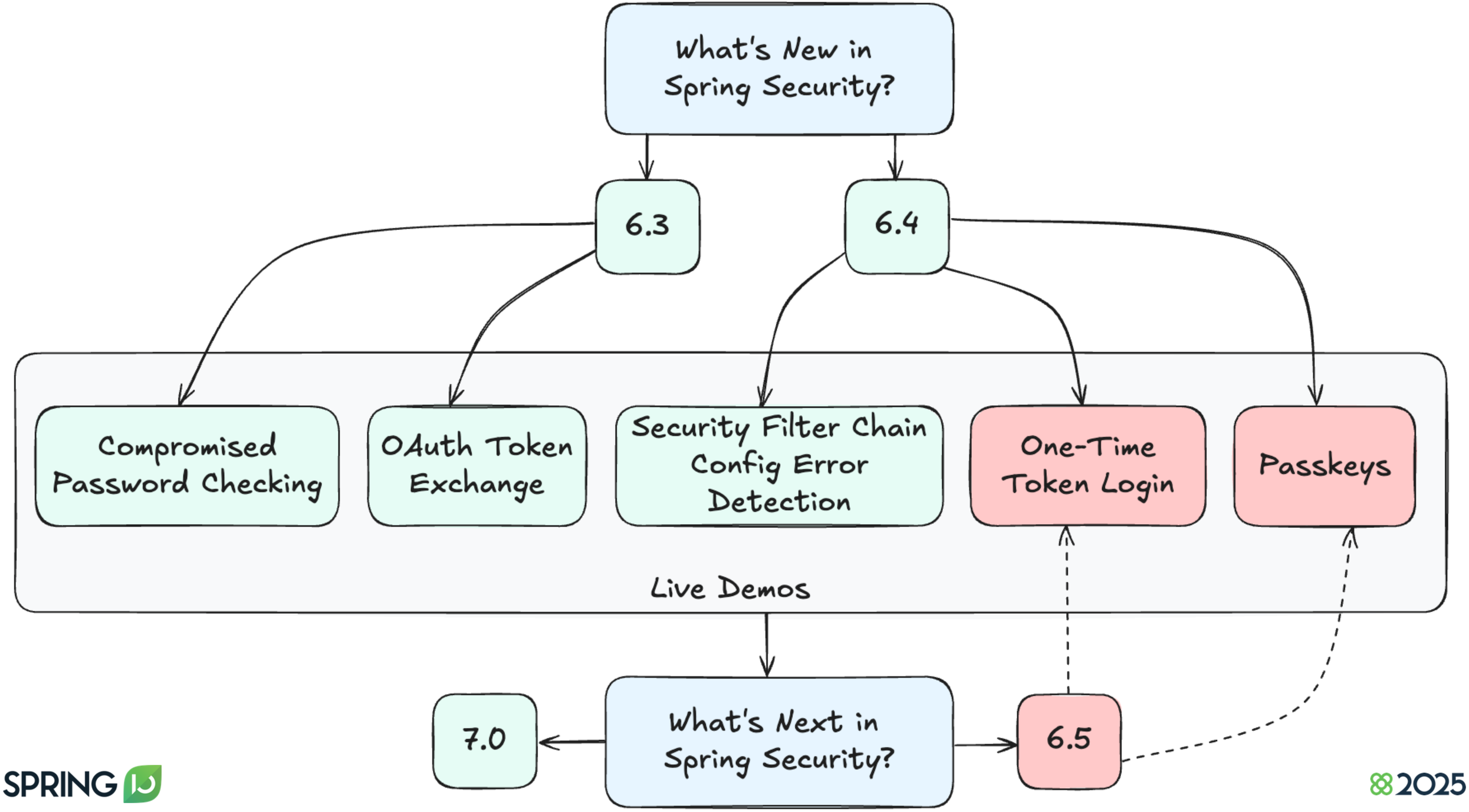
- Authentication
  - OAuth 2.0 Support for RestClient
  - OpenSAML 5 Support
- Authorization
  - Annotation templates support for @AuthenticationPrincipal and @CurrentSecurityContext
- Improved Kotlin Support (i.e., @Pre-/@PostFilter

```
@Target(TargetType.TYPE)
@Retention(RetentionPolicy.RUNTIME)
@AuthenticationPrincipal("claims['{claim}']")
@interface CurrentUsername {
    String claim() default "sub";
}

// ...

@GetMapping
public String method(@CurrentUsername("username") String username) {
    // ...
}
```

https://docs.spring.io/spring-security/reference/6.4/whats-new.html


What's New in Spring Security?
6.3
6.4
Compromised Password Checking
OAuth Token Exchange
Security Filter Chain Config Error Detection
One-Time Token Login
Passkeys
Live Demos
7.0
What's Next in Spring Security?
6.5

What's Next in Spring Security 6.5 (Just Released 19.5.2025) 😎

- Support for OAuth 2.0 Demonstrating Proof of Possession (RFC-9449 - DPoP)
- JDBC Persistence for WebAuthn/Passkeys
- Customizing One-Time Token Request
- Allow ***at+jwt*** for bearer tokens, according to RFC-9068 (JWT Profile for OAuth 2.0 Access Tokens)

https://docs.spring.io/spring-security/reference/6.5/whats-new.html
https://spring.io/blog/2025/05/19/spring-security-6-5-0-is-out
https://www.rfc-editor.org/rfc/rfc9449.html
https://www.rfc-editor.org/rfc/rfc9068.html

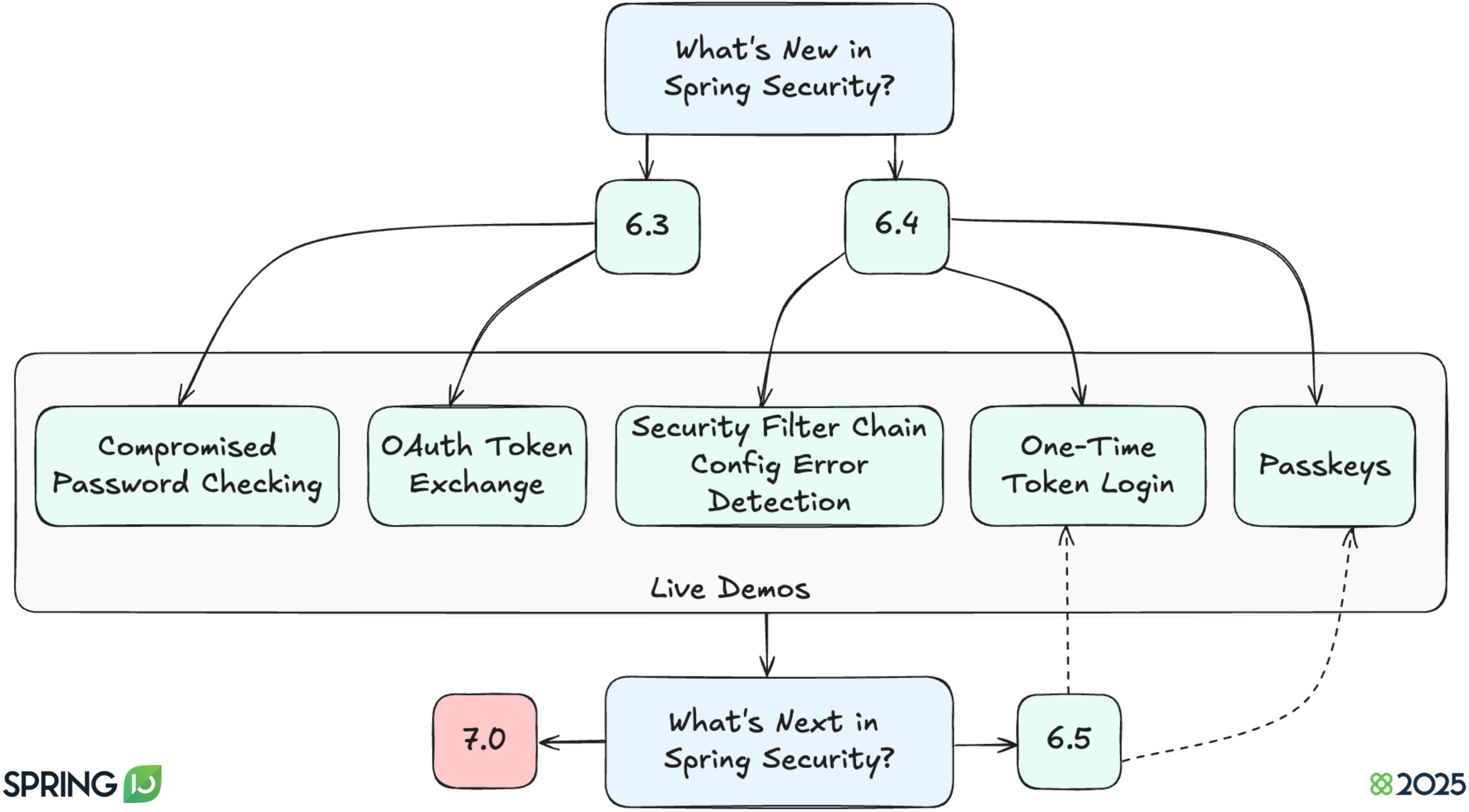
RFC 9449 - DPoP (Spring Security 6.5)

- Proof-of-Possesion Tokens (Token Binding)
  for Public Clients (i.e., a SPA)
  → Alternative to BFF pattern for SPA

- Prevent unauthorized parties from using leaked or stolen access tokens

- Requires an IdP capable of DPoP
  like Spring Authorization Server V.1.5+

https://www.rfc-editor.org/rfc/rfc9449.html
https://www.ietf.org/archive/id/draft-ietf-oauth-browser-based-apps-24.html#name-backend-for-frontend-bff


What's New in Spring Security?
6.3
6.4
Compromised Password Checking
OAuth Token Exchange
Security Filter Chain Config Error Detection
One-Time Token Login
Passkeys
Live Demos
What's Next in Spring Security?
7.0
6.5

What's Next in Spring Security 7.0?

- Breaking Changes !!!!
  - Mandatory use of Lambda DSL Configuration
  - Removal of Deprecated Code
- Enabling PKCE for Authorization Code by Default
- ...

```java
@Configuration
@EnableWebSecurity
public class SecurityConfig {

    @Bean
    public SecurityFilterChain filterChain(HttpSecurity http) throws Exception {
        http
            .authorizeHttpRequests(authorize -> authorize
                .requestMatchers("/blog/**").permitAll()
                .anyRequest().authenticated()
            )
            .formLogin(formLogin -> formLogin
                .loginPage("/login")
                .permitAll()
            )
            .rememberMe(Customizer.withDefaults());

        return http.build();
    }
}
```

https://docs.spring.io/spring-security/reference/6.5/migration-7/index.html
https://github.com/spring-projects/spring-security/milestone/270 (7.0.0-M1)

Bonus: What's New/Next in Spring Authorization Server

1.5.0

- OAuth 2.0 Pushed Authorization Requests (PAR)
- OAuth 2.0 Demonstrating Proof of Possession (DPoP)

1.3.0

- Mutual-TLS Client Certificate-Bound Access Tokens
- OAuth 2.0 Token Exchange
- Multi-Tenancy (Multiple Issuer)

1.4.0

- SPA sample using Backend For Frontend and Spring Cloud Gateway
- OpenID Connect 1.0 prompt=none

https://spring.io/blog/2025/05/20/spring-authorization-server-1-5-goes-ga
https://spring.io/blog/2024/11/19/spring-authorization-server-1-4-goes-ga
https://spring.io/blog/2024/05/22/spring-authorization-server-1-3-goes-ga

Join my Workshop later at 14:30 – 16:30 ☺

https://github.com/andifalk/springio25-security-workshop

WORKSHOPS 1 (ROOM 4)

Next-Gen Security with Spring: Passkeys, Token Exchange, and Authorization Enhancements [Workshop]

Andreas Falk

Workshop continuation

THANKS!!!

Andreas Falk

@andifalk

LinkedIn

https://www.linkedin.com/in/andifalk

https://github.com/andifalk/whats-new-in-spring-security