



# Spring Security

---

Beginner to Guru

Spring Security Authentication Events



## Spring Security Authentication Events

- Spring Security will send an Authentication Success or Authentication Failure Event with every Authentication Attempt
- These events are important hooks for monitoring system access
- Can be used for:
  - Logging who logged in when and from where
  - Failed log in attempts
  - Automatically lock accounts for too many attempts





## Spring Security Authentication Events

- Spring Security Authentication Events use the Event functionality found in Spring Framework
- Publish / Subscribe Type of Model
  - Spring Security will publish the event
  - One or more listeners can register to receive the event
- Available in all Spring Security Provided Authentication Providers
  - You are responsible for event publishing if using custom provider





## Default Event Mappings Authentication Events

- `BadCredentialsException` = `AuthenticationFailureBadCredentialsEvent`
- `UsernameNotFoundException` = `AuthenticationFailureBadCredentialsEvent`
- `AccountExpiredException` = `AuthenticationFailureExpiredEvent`
- `ProviderNotFoundException` = `AuthenticationFailureProviderNotFoundException`
- `DisabledException` = `AuthenticationFailureDisabledEvent`
- `LockedException` = `AuthenticationFailureLockedEvent`
- `AuthenticationServiceException` = `AuthenticationFailureServiceExceptionEvent`





## Default Event Mappings Authentication Events

- `CredentialsExpiredException` = `AuthenticationFailureCredentialsExpiredEvent`
- `InvalidBearerTokenException` = `AuthenticationFailureBadCredentialsEvent`



## Customization of Events

- Spring Security can be configured for additional or custom Authentication Events
  - Provide own instance of AuthenticationEventPublisher
    - Customize as needed



