

# Simplified Configuration

## Outline

You can simplify server security configurations in eGovFramework 3.0, using the pre-set configurations via XML Schema.

## Configuring XML Namespace and Schema

To simplify configuration, you'll need to declare XML as follows:

```
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:egov-security="http://www.egovframe.go.kr/schema/egov-security"
       xmlns:security="http://www.springframework.org/schema/security"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.2.xsd
http://www.egovframe.go.kr/schema/egov-security http://www.egovframe.go.kr/schema/egov-security/egov-security-3.0.xsd
http://www.springframework.org/schema/security
http://www.springframework.org/schema/security/spring-security-3.2.xsd">
```

## Security Configuration

Security Configuration information is provided as follows:

Example:

```
<egov-security:config id="securityConfig"
    loginUrl="/cvpl/EgovCvplLogin.do"
    logoutSuccessUrl="/cvpl/EgovCvplLogin.do"
    loginFailureUrl="/cvpl/EgovCvplLogin.do?login_error=1"
    accessDeniedUrl="/system/accessDenied.do"

    dataSource="dataSource"
    jdbcUsersByUsernameQuery="SELECT
USER_ID,PASSWORD,ENABLED,USER_NAME,BIRTH_DAY,SSN FROM USERS WHERE USER_ID = ?"
    jdbcAuthoritiesByUsernameQuery="SELECT USER_ID,AUTHORITY FROM AUTHORITIES WHERE
USER_ID = ?"
    jdbcMapClass="egovframework.rte.fdl.security.userdetails.EgovUserDetailsMapping"

    requestMatcherType="regex"
    hash="md5"
/>
```

## Description

Property	Description	Require	Remark
loginUrl	URL of log-in page	Require	
logoutSuccessUrl	Page URL called upon	Require	

	log-out
loginFailureUrl	URL of the page called upon log-in d failure
accessDeniedUrl	URL of the page called when access is denied
dataSource	dataSource for DBMS Optiona l configuration
jdbcUsersByUsernameQuery	Username query for authentication
jdbcAuthoritiesByUsername Query	Username query for authority check for the authenticate d users
jdbcMapClass	Class for user information mapping
requestMatcherType	Pattern Matching (regex, ant, ciRegex: 1 case-insensitive regex)
hash	Password encryption (sha-256, plaintext, sha, md5, bcrypt)
hashBase64	Use of base64 encoding for hash
concurrentMaxSessions	Maximum concurrent sessions
concurrentExpiredUrl	URL of the page redirected when attempting to access an expired URL

	URL of the page
defaultTargetUrl	redirected when log-in was successful
	Optional
	Redirects to the originally attempted URL when not assigned.

- DefaultMapUserDetailsMapping : Processes mapping without VO (You can generate Hash Map key by way of camel case method, for the designated column on jdbcUsersByUsernameQuery)

## Configuring Security Config Initializer

Refer to the following example for how to provide initialization information for security:

Example:

```
<egov-security:initializer id="initializer"
    supportPointcut="true"
/>
```

### Description

Property	Description	Required	Remark
supportPointcut	Select to support pointcut method	Optional	default : false
supportMethod	Select to support method	Optional	default : true

## Configuring Security Object Config

Basic query configuration information is provided as follows:

Example:

```
<egov-security:secured-object-config id="securedObjectConfig"
    roleHierarchyString="
        ROLE_ADMIN > ROLE_USER
        ROLE_USER > ROLE_RESTRICTED
        ROLE_RESTRICTED > IS_AUTHENTICATED_FULLY
        IS_AUTHENTICATED_FULLY > IS_AUTHENTICATED_REMEMBERED
        IS_AUTHENTICATED_REMEMBERED >
        IS_AUTHENTICATED_ANONYMOUSLY"
    sqlRolesAndUrl="
        SELECT auth.URL url, code.CODE_NM authority
        FROM RTETNAUTH auth, RTETCCODE code
        WHERE code.CODE_ID = auth.MNGR_SE"
/>
```

### Description

Property	Description	Required	Remark
roleHierarchyString	Assignment of hierarchy strings	Optional	Designates duly designated configuration information when not assigned
sqlRolesAndUrl	Query to designate URL-type role	Optional	Processes basic query of SecuredObjectDAO when not assigned
sqlRolesAndMethod	Query to designate method-type role	Optional	" "

sqlRolesAndPointcut	Query to designate pointcut-type role	Optional	"
sqlRegexMatchedRequestMapping	Query to designate the best matching URL for every request	Optional	"
sqlHierarchicalRoles	Query for hierarchical processing	Optional	"