

The 2024 Developer Survey results are live!

[See the results](#)

# Spring Boot 3 Keycloak Multi Tenant Configuration

Asked 1 year, 2 months ago Modified 1 year, 1 month ago Viewed 3k times



1



I have a Spring Boot 3 microservice and a standalone keycloak which has multiple realms. Users can call endpoints and pass the "Authorization" header there, which contains the Bearer jwt token, which specifies one of the existing realms.

How is it possible to implement this behavior when the spring boot microservice accesses the required realm to authenticate the current request, determining the realm dynamically based on the request?

Please note that I am using Spring boot v3.0.6. Previously worked based on the topic [Spring Boot Keycloak Multi Tenant Configuration](#) but now we can't use previous approach as was said there [Use Keycloak Spring Adapter with Spring Boot 3](#)

I have only one idea's create few oauth2 providers for each realm and somehow implement the choice for target provider in runtime. Thank you very much in advance!

java

spring-boot

spring-security

keycloak

Share Improve this question

Follow

edited Jun 23, 2023 at 9:47



Brian Clozel

58.5k ● 15 ● 174 ● 179

asked May 12, 2023 at 10:43



MrLebovsky

88 ● 1 ● 11

I'm curious, why didn't the post below help or even answer your question? – [ch4mp](#) May 17, 2023 at 0:54

sorry about wating, I'm gonna add another answer – [MrLebovsky](#) May 17, 2023 at 4:30

## 2 Answers

Sorted by: Highest score (default)



2



You can follow [my tutorials](#), all are multi-tenant. All configured for "static" multi-tenancy, [but one](#) which demoes conf for "dynamic" tenants.

Some are using just Spring Boot "official" starters and some are using thin wrappers around it. With the latest, you can configure a [resource server](#) to accept JWTs issued by as many realms as you want with almost 0 Java conf:

```
<properties>
  <com.c4-soft.springaddons.version>6.1.9</com.c4-soft.springaddons.version>
</properties>
```



```
<dependencies>
  <dependency>
    <groupId>com.c4-soft.springaddons</groupId>
    <artifactId>spring-addons-webmvc-jwt-resource-server</artifactId>
    <version>${com.c4-soft.springaddons.version}</version>
  </dependency>
</dependencies>
```

```
@Configuration
@EnableMethodSecurity
public static class WebSecurityConfig {
}
```

```
scheme: http
origins:
${scheme}://localhost:4200,${scheme}://localhost:8080,${scheme}://localhost:8100
auth-server: https://localhost:8443
```

```
com:
  c4-soft:
    springaddons:
      security:
        cors:
          - path: /solutions/**
            allowed-origins: ${origins}
        issuers:
          - location: ${auth-server}/realms/realm1
            username-claim: $.preferred_username
            authorities:
              - path: $.realm_access.roles
              - path: $.resource_access.*.roles
          - location: ${auth-server}/realms/realm2
            username-claim: $.preferred_username
            authorities:
              - path: $.realm_access.roles
              - path: $.resource_access.*.roles
          - location: ${auth-server}/realms/realm3
            username-claim: $.preferred_username
            authorities:
              - path: $.realm_access.roles
              - path: $.resource_access.*.roles
        permit-all:
          - /actuator/health/readiness
          - /actuator/health/liveness
          - /v3/api-docs/**
```

```
server:
  ssl:
    enabled: false
```

```
---
```

```
scheme: https
```

```
server:
  ssl:
    enabled: true
```

```
spring:
  config:
    activate:
      on-profile: ssl
```

If the realms are generated at runtime (after the resource servers are started), but you don't want to use "my" starters, then you'll have to refer to [Spring Security reference documentation](#).

Share Improve this answer Follow edited May 17, 2023 at 0:50

answered May 12, 2023 at 16:52



ch4mp

11k ● 6 ● 37 ● 66



-1



Finally I found the way to build multy-tenancy using Spring Boot:

<https://docs.spring.io/spring-security/reference/reactive/oauth2/resource-server/multitenancy.html>

Share Improve this answer Follow

answered May 17, 2023 at 4:29



MrLebovsky

88 ● 1 ● 11

What is different from what is explained in the tutorials linked above (those with just boot "official" starters like [this one](#))? How much more simple is it compared to the solution detailed in tmy answer (using "my" starters)? Last, the link you provide is the one at the end of my answer... – ch4mp May 17, 2023 at 4:41

Yep I didn't see your link bellow answer. Marked your answer as a solution. – MrLebovsky May 18, 2023 at 8:37