

Adding Config Server with Microservices

Config Server Project Dependencies

1. Config Server
2. Actuators

Steps to enable Config Server

1. @EnableConfigServer in Spring Application
2. Server Port :8071
3. Add following properties in configserver application properties file-
 - a. `spring.profile.active: git`
 - b. `spring.cloud.config.server.git.uri = "YOUR_GIT_URL"`
 - c. `spring.cloud.config.server.git.default-label = main`
 - d. `spring.cloud.config.server.git.timeout = 5`
 - e. `spring.cloud.config.server.git.clone-on-start = true`
 - f. `spring.cloud.config.server.git.force-pull = true`
4. Enable health related actuators in Config Server as follow :
 - a. `management.endpoints.web.exposure.include = "*"`
 - b. `management.health.readiness-state.enabled = true`
 - c. `management.health.liveness-state.enabled = true`
 - d. `management.endpoint.health.probes.enabled = true`

Steps to enable Config Client

1. Add Config Client dependency in all microservices
2. Add Dependency Management and Spring Cloud Version in all microservices
3. Add following properties in the application properties file-
 - a. `spring.config.import = "optional:configserver:http://localhost:8071"`
 - b. `spring.profile.active="qa/prod"`
 - i. Optional, Profiles can also set using Environment Variables / JVM Variables

Docker Compose Update:

1. Add new service for Config Server same as other microservices
2. Add environment to all the microservices as follows:

accounts:

environment:

SPRING_CONFIG_IMPORT: "configserver:http://**configserver**:8071"

SPRING_APPLICATION_NAME: "APP_NAME"

SPRING_PROFILES_ACTIVE: prod / default / qa

depends_on:

configserver:

condition: service-healthy

3. Define below configurations for all microservices to check config server status in Docker Compose YML file –

accounts.depends_on.configserver.condition = service-healthy

4. Add Health check Property in Docker Compose YML file–

**configserver.healthcheck.test = "curl -fail -silent
localhost:8071/actuator/health/readiness | grep UP | exit 1"**

configserver.healthcheck.timeout=5s

configserver.healthcheck.interval=10s

configserver.healthcheck.retries=10

configserver.healthcheck.start-period=10s

* **configserver** is the name of the defined service in docker compose YML file