

Spring Cloud Nedir?

Spring Cloud, JVM tabanlı uygulamayı bulutlarda hızlı bir şekilde dağıtmak için araçlar sağlayan açık kaynaklı bir kitaplık olarak tanımlanmakta. Dağıtılmış konfigürasyon, Devre kesiciler, Global kilitler, Servis kayıtları, Yük dengeleme, Küme durumu, Yönlendirme vb. gibi çeşitli özellikler sayesinde genişletilebilir bir mekanizma ve rakiplerine göre daha iyi bir kullanıcı deneyimi sağlamakta.

Spring Cloud Özellikleri

- 1- *Dağıtılmış yapılandırma(Distributed configuration)*
- 2- *Dağıtılmış mesajlaşma (Distributed messaging)*
- 3- *servisten servise aramalar (service-to-service calls)*
- 4- *Devre kesiciler(Circuit Breaker)*
- 5- *Küresel kilitler (Global locks)*
- 6- *Hizmet kaydı (Service registration)*
- 7- *Servis Keşfi (Service Discovery)*
- 8- *Yük dengeleme(Load balancing)*
- 9- *Küme durumu (Cluster state)*
- 10- *Yönlendirme(Routing)*

SPRING CLOUD COMPONENTS

Configuration:

Konfigürasyon, Spring Cloud için çok önemli bir bileşenidir, çünkü tüm dağıtılmış sistemlerde sunucu tarafı ile istemci tarafı arasındaki ilişkileri haricileştirilmiş konfigürasyon şeklinde sürdürür.

Service Discovery:

Yazılan mikro servislerin ayağa kalcıklarında kendilerini konfigüre edilen bir hub'a kayıt etmesi ve daha sonra bu hub' daki tüm servislerin birbirinden haberdar olarak iş yapmasına olanak vermektedir. Bu sayede tüm servis örneklerinin ip,port bilgilerini tutmak zorunda kalmıyoruz. Service discovery olarak, Consul, Zookeeper gibi opsiyonlar da sunuluyor aynı zamanda Netflix in geliştirmiş olduğu Eureka' yı kullanılıyor.

Circuit Breaker:

Devre/Adım/İşlem kesici

Devre/Adım/İşlem kesici. Devre kesiciler genellikle belirli durumlarda belirli servislere erişimi koruma altına almak için kullanılmakla beraber belirli/spesifik bir şekilde gelen bir istek işleme konulduğunda hata alınması kaçınılmaz ise yine bu durumda da devre kesici kullanılabilir. Ayrıca kullanıcıya bu duruma özel bir mesaj döndürülebilir ya da bu servise gelmeden önce yapması gereken bir işlem var ancak yapılmamış ise ilgili servislere yönlendirme yapılabilir.

Devre kesici genel olarak Netflix' in geliştirmiş olduğu Hystrix adlı aracı kullanılmaktadır. Hystrix de Spring Boot ile entegre etmesi, konfigüre etmesi ve kullanması oldukça kolay bir araçtır. Ayrıca Hystrix Dashboard adlı aracı kullanarak servislerimize gelen isteklere verdiğimiz cevapları ve bu cevapların sayılarını ve türlerini, response-time' larını görebiliriz.

Routing and Messaging:

Bu bileşenlerin her ikisi de Spring Cloud önemli bir yön ekler ve bu iletişimi sağlar. Yönlendirme Netflix Ribbon kullanır ve Feign(sahte istemci) açar, mesajlaşma Rabbit MQ veya Kafka'yı kullanır.

Load Balancing:

Mikroservislerimizin trafik dengelemesini izleyebilir, yönetebilir, dağıtabiliyoruz.

Round Robin algoritması ile çalışan ve kullanımı oldukça basit olan Ribbon adlı load-balancer' ı inceleyeceğiz.

Router and Filter:

Burada kullanacağımız araç yine Netflix' in geliştirmiş olduğu Zuul Proxy' dir. Zuul ile koşullar ve filtreler yazacağız, gelen istekleri diğer servislerimize zuul ile yönlendireceğiz.

API Gateway

API, Uygulama Programı Arayüzü anlamına gelir. Yazılım uygulamaları oluşturmak için bir dizi talimat, protokol ve araçtır. Yazılım bileşenlerinin nasıl etkileşime girmesi gerektiğini belirtir. API Ağ Geçidi bir sunucudur. Bir sisteme tek bir giriş noktasıdır. API Gateway, dahili sistem mimarisini kapsar. Her müşteriye göre uyarlanmış bir API sağlar. Ayrıca kimlik doğrulama, izleme, yük dengeleme, önbelleğe alma, istek şekillendirme ve yönetimi ve statik yanıt işleme gibi başka sorumlulukları da vardır.

API Gateway, istek yönlendirme, oluşturma ve protokol çevirisinden de sorumludur. İstemci tarafından yapılan tüm istekler API Ağ Geçidinden geçer. Bundan sonra, API Ağ Geçidi, istekleri uygun mikro servise yönlendirir.