**Tugas Teori Microservice**

|  |  |
| --- | --- |
| **Nama** | **: Farid Fadlizil Ikram** |
| **NIM** | **: 2301083013** |
| **Kelas** | **: TK 2A** |

**Soal :**

1. Jelaskan Tentang Server side Discovery

2. Jelaskan Tentang Client side Discovery

3. Jelaskan kekurangan dan kelebihan dari no 1 dan 2 diatas

4. Jelaskan tentang Service Registry

**Jawab :**

**1. Server-side Discovery**

Server-side discovery adalah pola arsitektur di mana layanan klien tidak langsung mencari lokasi layanan tujuan, tetapi mengandalkan service registry dan load balancer yang berjalan di sisi server untuk menemukan dan menghubungkan layanan yang tersedia.

**Cara kerja:**

* Klien mengirim permintaan ke load balancer atau service gateway.
* Load balancer mengambil daftar layanan yang terdaftar di service registry.
* Load balancer memilih instance layanan yang sesuai dan meneruskan permintaan ke layanan tersebut.

**Contoh Implementasi:**

* AWS Elastic Load Balancer (ELB)
* Kubernetes Service (k8s Service)
* Istio Service Mesh

**2. Client-side Discovery**

Client-side discovery adalah pola di mana klien bertanggung jawab langsung untuk menemukan layanan tujuan. Klien berkomunikasi dengan service registry untuk mendapatkan daftar instance yang tersedia dan memilih salah satu untuk berkomunikasi.

**Cara kerja:**

* Klien meminta daftar instance layanan dari service registry.
* Klien memilih satu instance berdasarkan strategi load balancing (misalnya round-robin atau least connections).
* Klien menghubungi layanan yang dipilih langsung tanpa perantara load balancer server.

**Contoh Implementasi:**

* Netflix Eureka
* Consul
* Zookeeper

**3. Kelebihan dan Kekurangan**

| **Metode** | **Kelebihan** | **Kekurangan** |
| --- | --- | --- |
| **Server-side Discovery** | - Klien lebih sederhana karena tidak perlu menangani discovery sendiri. - Load balancer di server dapat mengoptimalkan distribusi lalu lintas. - Lebih fleksibel untuk berbagai jenis klien (misal, aplikasi web dan mobile). | - Ada tambahan latensi karena perantara (load balancer). - Bergantung pada load balancer yang bisa menjadi bottleneck atau single point of failure. |
| **Client-side Discovery** | - Tidak ada latensi tambahan karena tidak menggunakan load balancer tambahan. - Mengurangi titik kegagalan tunggal (tidak ada load balancer server yang bisa gagal). | - Klien menjadi lebih kompleks karena harus menangani discovery dan load balancing sendiri. - Setiap klien harus dikonfigurasi dengan service registry yang benar. |

**4. Service Registry**

Service registry adalah komponen yang menyimpan informasi tentang instance layanan yang tersedia dalam suatu sistem terdistribusi. Registry ini digunakan dalam pola discovery untuk memungkinkan layanan saling menemukan.

**Cara kerja:**

1. Setiap layanan yang tersedia mendaftarkan dirinya ke service registry.
2. Ketika layanan baru muncul atau mati, registry diperbarui secara otomatis.
3. Klien atau load balancer menggunakan registry ini untuk menemukan layanan yang sesuai.

**Contoh Service Registry:**

* **Eureka (Netflix)**: Banyak digunakan dalam sistem berbasis microservices.
* **Consul (HashiCorp)**: Selain registry, juga menyediakan fitur health check.
* **Zookeeper (Apache)**: Digunakan untuk koordinasi layanan dalam sistem terdistribusi.