1. Jelaskan Tentang Server side Discovery

2. Jelaskan Tentang Cleint side Discovery

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

4. Jelaskan tentang Service Registry  
  
**Jawaban:**

1. **Server-Side Discovery**

Dalam pola ini, load balancer atau gateway bertindak sebagai perantara yang mengelola routing permintaan klien. Klien hanya mengirim permintaan ke alamat load balancer (seperti AWS Elastic Load Balancing atau Kubernetes Service), lalu komponen server ini akan:

1. Mengkueri Service Registry untuk menemukan instansi layanan yang tersedia
2. Memilih instansi yang sehat berdasarkan mekanisme load balancing
3. Meneruskan permintaan ke instansi yang dipilih

Contoh implementasi: API Gateway dengan NGINX atau solusi cloud seperti AWS ALB.

1. **Client-Side Discovery**

Pola ini memberikan tanggung jawab discovery ke klien. Klien akan:

1. Langsung berkomunikasi dengan Service Registry (seperti Consul atau Eureka) untuk mendapatkan daftar instansi layanan
2. Memilih instansi menggunakan algoritma load balancing client-side
3. Mengirim permintaan langsung ke instansi yang dipilih

Contoh implementasi: Netflix Ribbon dengan Eureka Service Registry.

1. **Kelebihan dan Kekurangan**

|  |  |  |
| --- | --- | --- |
| **Aspek** | **Server-Side Discovery** | **Client-Side Discovery** |
| **Kompleksitas Klien** | Lebih sederhana (tidak perlu logika discovery) | Lebih kompleks (harus implementasi load balancing) |
| **Skalabilitas** | Bergantung kapasitas load balancer | Lebih skalabel (tidak ada bottleneck sentral) |
| **Biaya** | Mahal (perlu infrastruktur load balancer) | Lebih hemat biaya infrastruktur |
| **Latensi** | Tambahan hop jaringan ke load balancer | Lebih cepat (komunikasi langsung) |
| **Ketergantungan** | Tidak perlu integrasi registry di klien | Klien harus terintegrasi dengan registry |

1. **Service Registry**

Komponen sentral yang berfungsi sebagai database dinamis untuk melacak lokasi instansi layanan, dengan karakteristik:

* Registrasi Otomatis: Instansi layanan mendaftarkan diri saat startup dan menghapus registrasi saat shutdown.
* Health Checking: Memantau kesehatan instansi layanan secara berkala.
* Penyimpanan Metadata: Menyimpan informasi seperti versi, protokol, dan endpoint layanan.

Contoh implementasi: Consul, Netflix Eureka, atau Azure Spring Apps Service Registry. Service Registry menjadi fondasi penting baik untuk server-side maupun client-side discovery.