

Nama : Rayyan Nianto

NIM : 21107013

Prodi : Teknik Informatika

Buatlah 3 studi kasus tentang RPC beserta cara penyelesaiannya?

Jawab :

1. Studi Kasus: Sistem Pemesanan Tiket Online

Masalah: Sebuah perusahaan ingin mengembangkan sistem pemesanan tiket online yang diakses oleh pengguna melalui aplikasi mobile dan website. Sistem ini terhubung dengan server pusat yang menyimpan informasi terkait jadwal, harga, dan pemesanan tiket. Pengguna harus bisa memesan tiket secara real-time, dan data harus diperbarui di server agar dapat diakses oleh semua pengguna.

Solusi dengan RPC:

- Langkah 1: Implementasikan server RPC yang bertanggung jawab untuk mengelola jadwal tiket, harga, dan pemesanan.
- Langkah 2: Setiap aplikasi client (mobile atau web) akan melakukan panggilan RPC ke server saat pengguna mencari atau memesan tiket.
- Langkah 3: Server akan memproses permintaan dengan memeriksa ketersediaan tiket, melakukan pemesanan, dan memperbarui database.
- Langkah 4: Server RPC mengembalikan hasil (sukses atau gagal) ke client, yang kemudian ditampilkan ke pengguna.

Penyelesaian:

- Server dan client berkomunikasi melalui protokol seperti gRPC atau XML-RPC.
- Server melakukan sinkronisasi data secara real-time, memastikan bahwa pengguna tidak memesan tiket yang sama dalam waktu bersamaan.
- Masalah seperti latensi dapat diatasi dengan penggunaan caching dan load balancing.

2. Studi Kasus: Sistem Monitoring Jarak Jauh untuk Jaringan

Masalah: Sebuah perusahaan ingin mengimplementasikan sistem monitoring jarak jauh yang memantau status server dan perangkat jaringan di beberapa lokasi. Administrator jaringan harus bisa mengirim perintah seperti reboot server, cek status CPU, dan membaca log dari komputer client jarak jauh.

Solusi dengan RPC:

- Langkah 1: Setiap perangkat jaringan memiliki RPC server yang berjalan di dalamnya untuk menerima perintah dari server pusat.

- Langkah 2: Administrator menggunakan aplikasi client untuk mengirimkan permintaan RPC (misalnya, reboot server, cek status CPU) ke perangkat jarak jauh.
- Langkah 3: Server RPC pada perangkat jarak jauh menerima permintaan, menjalankan perintah, dan mengirim kembali hasil ke client.

Penyelesaian:

- Gunakan protokol RPC seperti JSON-RPC atau gRPC untuk mengirim perintah ke perangkat jarak jauh.
- Server di sisi perangkat menangani tugas dengan cara yang aman, termasuk autentikasi dan otorisasi untuk memastikan hanya pengguna yang diizinkan yang dapat mengakses fungsi tersebut.
- Sistem juga dapat disertai dengan mekanisme logging untuk memantau semua aktivitas yang dilakukan melalui RPC.

3. Studi Kasus: Aplikasi Perbankan Terdistribusi

Masalah: Sebuah bank ingin mengembangkan aplikasi perbankan yang memungkinkan nasabah melakukan transaksi seperti transfer dana, cek saldo, dan pembayaran tagihan melalui berbagai perangkat (mobile, web, ATM). Aplikasi ini harus aman dan mampu menangani volume transaksi yang tinggi.

Solusi dengan RPC:

- Langkah 1: Implementasikan backend RPC server yang mengelola transaksi dan mengotentikasi pengguna.
- Langkah 2: Setiap kali nasabah melakukan tindakan seperti cek saldo atau transfer dana, aplikasi client akan mengirimkan permintaan RPC ke server.
- Langkah 3: Server akan memproses permintaan dengan memastikan bahwa pengguna memiliki otorisasi yang sesuai, melakukan transaksi, memperbarui database, dan mengirim hasil kembali ke client.

Penyelesaian:

- Gunakan protokol RPC seperti gRPC dengan enkripsi (TLS) untuk memastikan keamanan data saat dikirim antara client dan server.
- Backend harus disesuaikan dengan sistem terdistribusi untuk menghindari bottleneck dan downtime, misalnya dengan menggunakan load balancing dan replikasi database.
- Sistem harus mendukung penanganan kegagalan transaksi dengan memastikan bahwa jika terjadi kesalahan (misalnya jaringan terputus), transaksi bisa diulang atau dibatalkan secara aman.