

# SOAL 11

## 1. Analisis Query dengan Penggunaan CPU Tinggi

Pada sebuah server MySQL dengan jumlah user yang tinggi, CPU usage meningkat drastis saat melakukan beberapa query tertentu. Sebutkan langkah-langkah yang dapat Anda lakukan untuk:

1. Mengidentifikasi query mana yang menyebabkan penggunaan CPU tinggi.
2. Menyajikan laporan performa query tersebut menggunakan tools monitoring MySQL seperti EXPLAIN dan SHOW PROCESSLIST.
3. Berikan solusi apa yang dapat Anda terapkan untuk mengoptimalkan query tersebut agar penggunaan CPU lebih efisien.

## JAWABAN

### 1. Mengidentifikasi Query Penyebab Penggunaan CPU Tinggi

- **Monitor Slow Query Log:** Aktifkan `slow_query_log` di MySQL untuk mencatat query yang memerlukan waktu lebih lama dari nilai `long_query_time` yang ditetapkan. Slow Query Log berguna untuk mendeteksi query dengan eksekusi lambat yang dapat menyebabkan penggunaan CPU berlebih.

#### CONTOH QUERY:

```
SET GLOBAL slow_query_log = 1;
```

```
SET GLOBAL long_query_time = 1; -- Misalnya, 1 detik
```

### 2. Menganalisis Performa Query dengan EXPLAIN

- **Gunakan SHOW PROCESSLIST:** Perintah ini menunjukkan semua query yang sedang berjalan pada MySQL server. Cari query yang menempati waktu eksekusi tinggi (Time) dan memiliki status seperti Copying to tmp table, Sorting, atau Sending data, yang bisa mengindikasikan penggunaan CPU yang intensif.

#### CONTOH QUERY:

```
SHOW PROCESSLIST;
```

- **Gunakan EXPLAIN:** Jalankan EXPLAIN pada query yang sudah diidentifikasi menyebabkan beban CPU tinggi. EXPLAIN membantu mengetahui cara MySQL memproses query, tabel apa yang di-scan, indeks apa yang digunakan, serta apakah ada operasi seperti JOIN, Sorting, atau Temporary Table.

#### CONTOH QUERY (Case di tugas ini):

```
EXPLAIN SELECT p.id, p.content, p.created_at, u.username
FROM posts p
JOIN follows f ON p.user_id = f.following_id
JOIN users u ON p.user_id = u.id
WHERE f.follower_id = 1
ORDER BY p.created_at DESC
LIMIT 20 OFFSET 0;
```

### 3. Solusi Optimalisasi Query

- **Indexing:** Pastikan query menggunakan indeks pada kolom yang sering muncul di kondisi WHERE, JOIN, ORDER BY, dan GROUP BY.
- **Optimalkan JOIN:** Gunakan JOIN hanya jika diperlukan dan pastikan kolom yang di-JOIN memiliki indeks.
- **Gunakan LIMIT:** Pada query yang tidak membutuhkan seluruh data, gunakan LIMIT untuk membatasi jumlah data yang diambil.
- **Cache Query Hasil (Query Cache):** Aktifkan MySQL Query Cache untuk query yang sering diulang jika memungkinkan.