

**LAPORAN PRAKTIKUM
JARINGAN KOMPUTER
TRANSPORT LAYER PROTOCOL**



Dosen : Jauari Akhmad Nur Hasim S.ST., M.Kom.

Disusun oleh:

Dinda Amalia FM

5122500003

2 D3 MMB A

**PRODI D3 TEKNIK MULTIMEDIA BROADCASTING
TEKNOLOGI MULTIMEDIA KREATIF
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
2023/2024**

BAB I

PENDAHULUAN

1.1 Tujuan

- Mahasiswa mampu memahami bagaimana cara kerja Transport Layer Protocol.
- Mahasiswa mampu memahami perbedaan antara TCP dan UDP.

1.2 Dasar Teori

Dalam model referensi OSI, lapisan yang keempat adalah Transport Layer Protocol. Lapisan ini memiliki peran krusial dalam proses pengiriman data antara dua perangkat yang berkomunikasi di jaringan komputer. Terdapat dua protokol yang umum digunakan di lapisan ini adalah TCP (Transmission Control Protocol) dan UDP (User Datagram Protocol).

Yang pertama yaitu TCP. Sebagai protokol yang dapat diandalkan, TCP memastikan bahwa data dikirim dengan benar dan lengkap antara pengirim dan penerima melalui mekanisme pemeriksaan kesalahan (checksum) dan pengaturan ulang (retransmission). Selain itu, TCP bertanggung jawab untuk mengatur aliran data agar sesuai dengan kapasitas jaringan dan mencegah overloading.

Kelebihan TCP

- Membantu untuk membuat/mengatur koneksi antara berbagai jenis komputer.
- Beroperasi secara independen dari sistem operasi.
- Mendukung banyak protokol routing
- Model TCP/IP memiliki arsitektur client-server yang sangat skalabel.

Kekurangan TCP

- Tidak pernah menyimpulkan transmisi tanpa semua data yang bergerak diminta secara eksplisit.
- Tidak cocok digunakan untuk transmisi broadcast atau multicast.
- Boros bandwidth, waktu, atau tenaga dikarenakan memiliki banyak fitur yang mungkin tidak diperlukan.

Sebaliknya, UDP adalah protokol yang lebih sederhana dan lebih cepat. Meskipun tidak menjamin pengiriman data yang handal seperti TCP, UDP sangat cocok untuk aplikasi yang membutuhkan transmisi data tanpa delay yang signifikan. Aplikasi seperti streaming video dan voice over IP (VoIP) sering menggunakan UDP karena fokusnya lebih pada kecepatan transmisi daripada integritas data yang sempurna.

Kelebihan UDP

- Transmisi yang lebih cepat karena tanpa mekanisme pengaturan ulang.
- Overhead lebih rendah, memberikan kinerja tinggi.
- Mendukung pengiriman data ke beberapa penerima secara efisien. (Broadcast & Multicast)

Kekurangan UDP

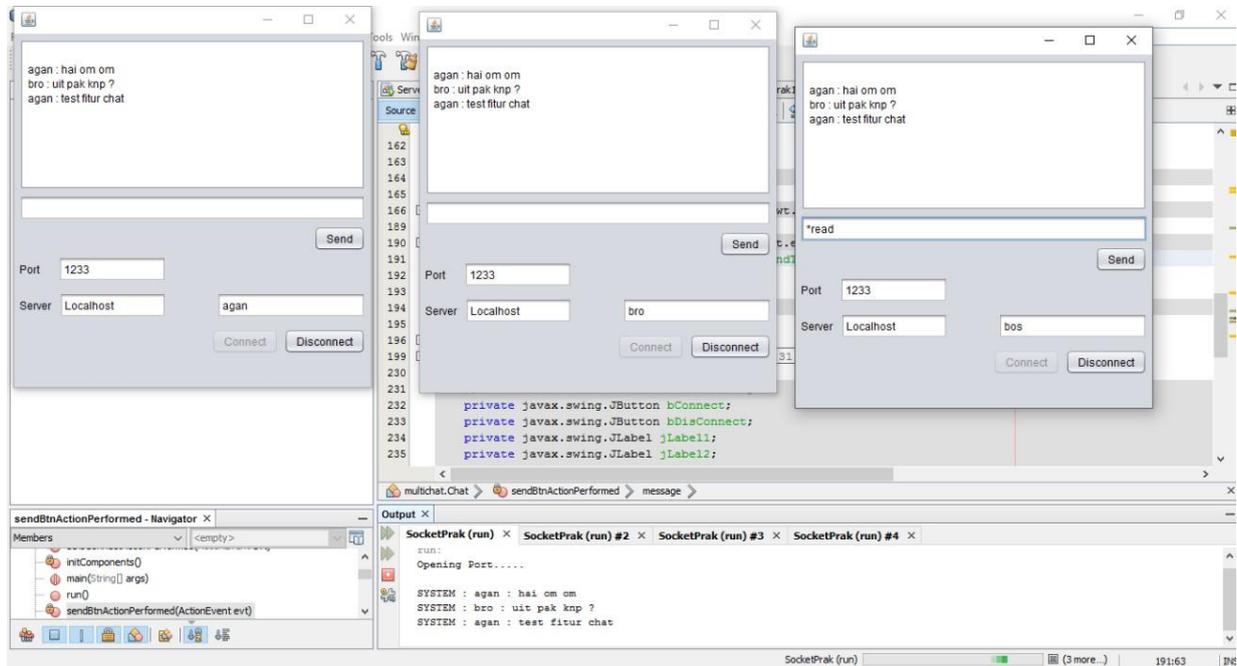
- Tidak menjamin pengiriman data
- Tidak menjamin urutan data yang sama saat diterima.
- Kurang cocok untuk aplikasi yang memerlukan keandalan dan toleransi kesalahan tinggi.

Baik TCP maupun UDP memiliki kelebihan dan kekurangan, dan pilihan antara keduanya tergantung pada aplikasi atau layanan yang berjalan di atasnya.

BAB II PEMBAHASAN

2.1 Hasil Pembahasan

Percobaan Aplikasi Multichat Chat.Java



Analisa Aplikasi Multichat Chat.Java

Jika dilihat dari listing programnya, aplikasi multichat ini termasuk ke dalam TCP. Hal ini dikarenakan class `cThread` yang digunakan adalah kode “Socket” untuk berkomunikasi dengan client nya.

```
// set in
// set out
host = InetAddress.getLocalHost();
link = new Socket(address:host, port:Server.PORT);
tfIP.setText(t: String.valueOf(t: Server.PORT));
tfIP.setText(t: "localhost");

name = tfName.getText();
```

Di sisi lain juga terdapat kode “BufferedReader” & “PrintWriter” yang digunakan untuk membaca dari dan menulis ke socket, yang umumnya digunakan dalam komunikasi TCP.

```
client = socket;
try {
    in = new BufferedReader(new InputStreamReader(in: client.getInputStream()));
    out = new PrintWriter(out:client.getOutputStream(), autoFlush: true);
} catch (IOException e) {
    e.printStackTrace();
}
```

Serta saat pesan diterima dari klien (received) diambil dari “in” dengan “in.readLine()”, yang sesuai dengan protokol TCP yang mengirimkan pesan dalam bentuk aliran teks yang dapat dibaca menggunakan “BufferedReader”.

```
@Override
public void run() {
    String response = "";
    do {
        try {
            response = in.readLine();
            String resp = jTextField1.getText()+"\n"+response;
            jTextField1.setText(resp);
        } catch (IOException ex) {
            Logger.getLogger(ChatClient.class.getName()).log(Level.SEVERE, msg:null,
            ex);
        }
    } while (!response.equals("QUIT"));
}
```

Jadi, kode-kode diatas adalah implementasi TCP untuk handle koneksi dengan klien dalam aplikasi multichat.

BAB III PENUTUP

3.1 Kesimpulan

Pemilihan antara TCP dan UDP tergantung pada kebutuhan aplikasi dan karakteristik data yang dikirim. TCP cocok untuk aplikasi yang memerlukan keandalan dan urutan data yang ketat, sementara UDP lebih sesuai untuk aplikasi yang memprioritaskan kecepatan dan responsivitas. Keputusan ini bergantung pada trade-off antara overhead dan keandalan yang diinginkan dalam situasi tertentu.