

## JARINGAN KOMPUTER – MODUL 8

### TOPIK TUGAS BESAR

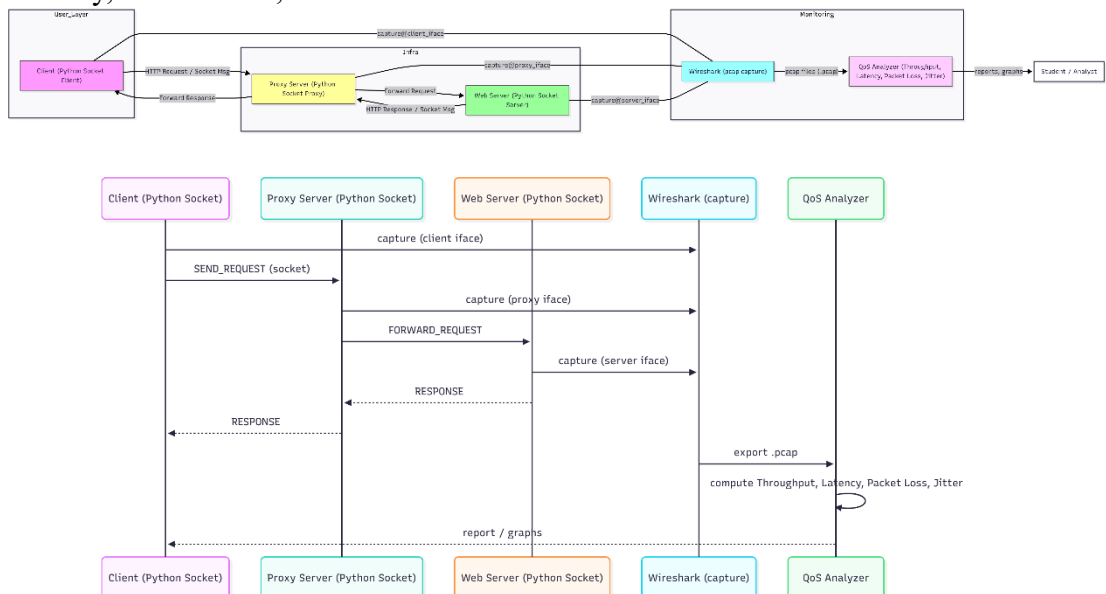
#### IMPLEMENTASI PROXY SERVER, SOCKET PROGRAMMING, DAN ANALISIS QOS MENGGUNAKAN WIRESHARK

UNTUK MEMBANTU TUGAS BESAR BERIKUT DIBERIKAN LINK MATERI BERKAITAN: [sok-py-prog](#) dan [serweb-py-prog](#)

##### A. Latar Belakang (Deskripsi Tugas)

Jaringan komputer saat ini sangat bergantung pada pengelolaan dan pengukuran kualitas komunikasi antar perangkat. Salah satu cara untuk mengelola komunikasi tersebut adalah dengan menggunakan Proxy Server, yang bertindak sebagai perantara antara Client dan Web Server. Untuk memantau performa dan kualitas jaringan, digunakan alat analisis seperti Wireshark, yang dapat mengukur Quality of Service (QoS).

Pada tugas besar ini, mahasiswa akan mengimplementasikan Web Server, Proxy Server, dan Client menggunakan Socket Programming dengan bahasa Python. Selain itu, mahasiswa juga akan melakukan pengukuran performa jaringan menggunakan Wireshark, serta menganalisis parameter QoS seperti Throughput, Latency, Packet Loss, dan Jitter.



##### B. Tujuan (Target)

Setelah menyelesaikan tugas ini, mahasiswa diharapkan mampu:

1. Mengimplementasikan Socket Programming untuk komunikasi antara Web Server, Proxy Server, dan Client dengan menggunakan bahasa pemrograman Python.
2. Menggunakan Wireshark untuk menangkap dan menganalisis paket jaringan.
3. Mengukur Quality of Service (QoS) jaringan (throughput, latency, jitter, dan packet loss).
4. Menyusun laporan teknis yang menggambarkan implementasi, pengujian, dan analisis QoS.
5. Melakukan Coding On The Spot (COTS) di depan penguji untuk menunjukkan pemahaman terhadap implementasi.

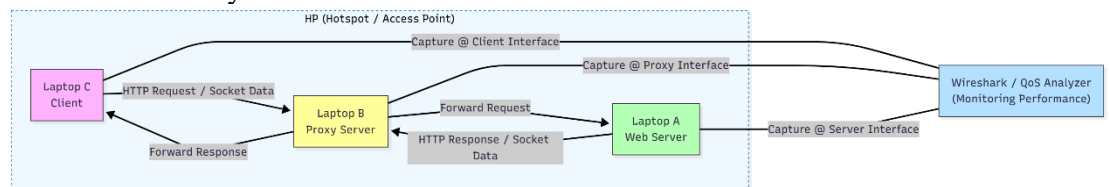
C. Tools yang digunakan

1. Laptop (sesuai dengan jumlah anggota kelompok)
2. WiFi hotspot (HP) atau router
3. Wireshark
4. Bahasa Pemrograman Python

D. Topologi dan Pembagian Peran Laptop

1. Kelompok dengan 3 Anggota (Utama)
  - Laptop A berperan sebagai Web Server
  - Laptop B berperan sebagai Proxy Server + Menyalakan Wireshark
  - Laptop C berperan sebagai Client

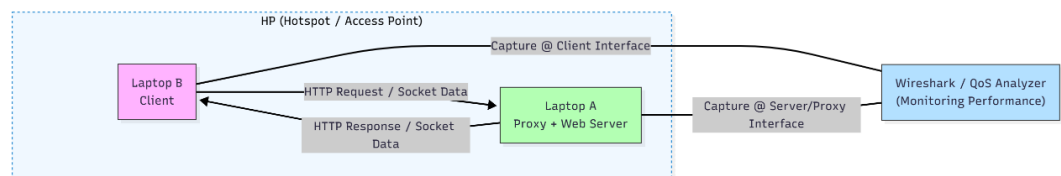
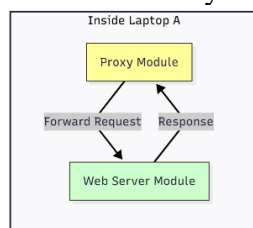
Berikut ilustrasinya



2. Kelompok dengan 2 Anggota (Kondisi Khusus)

- Laptop A berperan sebagai Web Server + Proxy Server + Menyalakan Wireshark
- Laptop B berperan sebagai Client

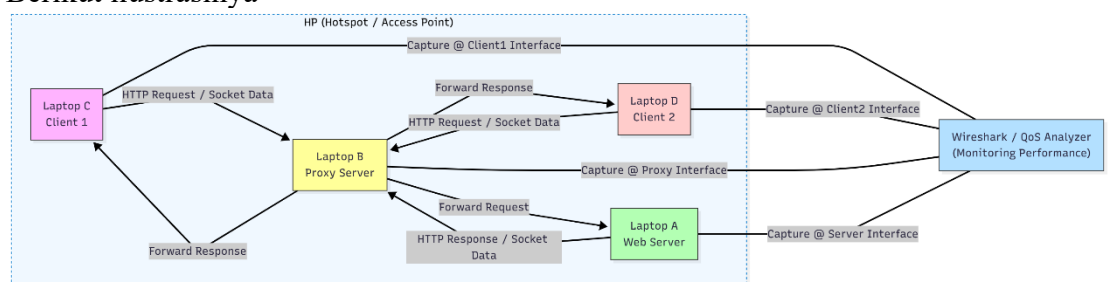
Berikut ilustrasinya



3. Kelompok dengan 4 Anggota (Kondisi Khusus)

- Laptop A berperan sebagai Web Server
- Laptop B berperan sebagai Proxy Server + Menyalakan Wireshark
- Laptop C berperan sebagai Client 1
- Laptop D berperan sebagai Client 2

Berikut ilustrasinya



E. Ketentuan saat Implementasi

a) Setup Jaringan

1. Aktifkan hotspot (tethering HP atau router)
2. Hubungkan semua laptop ke jaringan yang sama
3. Sinkronkan waktu antar Laptop (manual atau NTP)

b) Socket Programming

1. Web Server

Web Server berfungsi sebagai penyedia layanan (resource) untuk Client melalui Proxy. Server ini melayani permintaan data dan mengirimkan respon sesuai protokol yang digunakan.

Persyaratan Teknis:

- Menggunakan single-thread pada bagian acceptor untuk menerima koneksi TCP.
- Menggunakan multi-thread pada bagian worker pool untuk memproses beberapa request secara bersamaan.
- Menangani HTTP request dari Proxy atau Client melalui TCP port 8000.
- Mengirimkan HTTP response berupa halaman HTML sederhana, misalnya index.html, atau file statis lain seperti gambar dan CSS.
- Menyediakan layanan UDP pada port 9000 sebagai echo server untuk pengujian latency, jitter, dan packet loss.
- Menampilkan log koneksi pada terminal, meliputi: alamat IP client, waktu permintaan, file yang diminta, ukuran data, dan lama pemrosesan.
- Dapat dijalankan dalam dua mode:
  - Mode Single-Thread: seluruh request ditangani secara berurutan (untuk pengujian dasar).
  - Mode Threaded: setiap request diproses oleh thread terpisah (untuk uji performa dan QoS).
- Dilengkapi dengan timeout untuk mencegah koneksi menggantung dan logging otomatis untuk setiap aktivitas.

2. Proxy Server

Proxy Server bertugas sebagai perantara antara Client dan Web Server. Komponen ini bertanggung jawab meneruskan request, mengatur cache, serta mengoptimalkan aliran data agar Client dapat mengakses layanan dengan efisien.

Persyaratan Teknis:

- Menggunakan multi-threading agar mampu menangani banyak Client secara bersamaan.
- Beroperasi menggunakan dua protokol komunikasi:
  - TCP port 8080 untuk meneruskan permintaan HTTP dari Client ke Web Server.
  - UDP port 9090 untuk meneruskan datagram uji QoS antara Client dan Server.
- Menyediakan mekanisme cache sederhana untuk mempercepat permintaan yang sama (dengan status HIT atau MISS).
- Menerapkan connection timeout (5–10 detik) untuk menghindari koneksi tidak aktif.
- Menangani kesalahan jaringan dengan mengirimkan pesan error standar seperti:
  - HTTP/1.1 504 Gateway Timeout atau HTTP/1.1 502 Bad Gateway.
- Pada mode UDP, Proxy hanya meneruskan paket tanpa melakukan retransmission agar packet loss dan jitter dapat diukur secara akurat.
- Menampilkan log aktivitas transaksi, yang mencakup:
  - Alamat IP sumber dan tujuan,
  - Jenis protokol (TCP/UDP),
  - Status cache,
  - Ukuran data yang diteruskan,
  - Waktu proses total.

- Dapat dijalankan dengan konfigurasi thread pool untuk membatasi jumlah maksimum koneksi aktif.
3. Client
- Client bertugas untuk mengirimkan permintaan layanan (request) ke Proxy atau langsung ke Web Server, serta menerima respon yang diberikan. Selain itu, Client juga berperan dalam melakukan pengujian performa jaringan (QoS).
- Persyaratan Teknis:
- Dapat bekerja dalam dua mode komunikasi utama:
    - Mode TCP (HTTP): mengirimkan permintaan halaman web dan menerima konten HTML dari Web Server.
    - Mode UDP (Test QoS): mengirimkan sejumlah paket data ke Proxy atau Server untuk mengukur latency, jitter, dan packet loss.
  - Client dapat dijalankan dalam tiga kemungkinan pengujian:
    - Single Client: hanya satu Client yang aktif. Digunakan untuk pengujian dasar konektivitas dan fungsi sistem.
    - Multi-Client: beberapa Client dijalankan secara bersamaan (paralel) untuk simulasi beban jaringan **buatkan menjadi 5 client sekaligus**.
    - Menjalankan HTML (Browser Mode): Client menampilkan hasil respon HTML dari Server di jendela browser lokal (menggunakan HTTP response yang diterima).
  - Client menghitung dan menampilkan hasil pengujian, meliputi:
    - Throughput (kecepatan transfer data),
    - Latency rata-rata,
    - Packet Loss (% paket hilang),
    - Jitter (variasi waktu antar paket).
  - Hasil pengujian dapat disimpan dalam format log file atau CSV untuk dibandingkan dengan Wireshark
  - Menyediakan fitur konfigurasi:
    - IP dan port tujuan (Proxy/Server),
    - Jumlah paket (untuk mode UDP),
    - Ukuran paket,
    - Interval pengiriman antar paket.

## F. Skenario Pengujian dan Analisis QoS

### 1. Langkah Pengujian

- Pastikan semua laptop (Client, Proxy, dan Web Server) terhubung pada jaringan WiFi yang sama.
- Jalankan program:
  - Web Server di port 8000 (TCP) dan 9000 (UDP).
  - Proxy Server di port 8080 (TCP) dan 9090 (UDP).
- Jalankan Client dan lakukan semua jenis pengujian yang mungkin
- Buka Wireshark pada masing-masing laptop dan gunakan filter
- Lakukan tiga pengujian:
  - **Single Client** → satu koneksi.
  - **Multi-Client** → beberapa koneksi bersamaan.
  - **Menjalankan Browser**
- Simpan hasil capture dalam format **.pcap** untuk analisis.

### 2. Analisis QoS

Lakukan analisis QoS untuk setiap pengujian Client yang dilakukan yang terdiri dari: Throughput, Latency, Packet Loss dan Jitter

G. Pengumpulan File

Berikut merupakan file-file yang harus anda kumpulkan setelah melaksanakan berbagai aktivitas:

1. Kode semua program (dalam .zip)
2. Laporan Akhir dalam format PDF dengan struktur:
  - Cover (Judul, Nama dan NIM anggota kelompok, Nama dosen pengampu, Mata kuliah, logo universitas, program studi dan universitas, tanggal pengumpulan)
  - Penjelasan Pembagian Anggota dan File yang dikerjakan (**SETIAP ANGGOTA MINIMAL WAJIB MEMBUAT SATU FILE**)
  - Implementasi Socket Programming
    - Memberikan code
    - Penjelasan keterhubungan antara web server, proxy server, dan client
  - Hasil dan Pembahasan:
    - Hasil pengujian dan Perbandingan performa antar protocol
    - Diskusi hasil dan kendala pengujian
  - Kesimpulan dan Saran
  - Referensi
3. Slide Presentasi dalam format PDF dengan struktur:
  - Slide pertama merupakan Cover (Judul, Nama dan NIM anggota kelompok, Nama dosen pengampu, Mata kuliah, logo universitas, program studi dan universitas, tanggal pengumpulan)
  - Slide kedua – keenam merupakan hasil running (sambil dipraktikkan) dan hasil analisis Wireshark dan disertai perhitungan QoS-nya
  - Slide ketujuh merupakan penutup