

## ABSTRAK

Seiring berkembangnya kebutuhan akan teknologi, permintaan layanan data yang cepat dan efisien akan terus meningkat. Sistem *Multi Edge Caching* merupakan solusi untuk mengatasi permasalahan ini dengan menyimpan konten secara lokal di tepi jaringan yang memungkinkan untuk lebih dekat dengan pengguna akhir. Dengan mengurangi latensi dan mengurangi beban pada *server* utama, *multi edge caching* memungkinkan distribusi konten yang lebih cepat dan hemat sumber daya. *Apache Traffic Server* diimplementasikan sebagai *proxy server* dalam penelitian ini yang dapat meningkatkan efisiensi dan kinerja jaringan dengan menyimpan konten yang sering diminta di tepi jaringan. Penelitian ini mengkaji performa sistem dari segi kecepatan akses data, penghematan *bandwidth*, dan pengurangan latensi.

Untuk memudahkan dalam *monitoring* performa sistem, penelitian ini menggunakan *log dashboard* yang dibuat dengan *framework Flask* dan bahasa pemrograman *Python* lalu pengambilan data dari beberapa *Apache Traffic Server* melalui API dan menampilkan data tersebut dalam bentuk JSON. Penelitian ini menganalisis implementasi *multi edge caching* dalam konteks *website* multimedia dan layanan *Video on Demand* dengan fokus pada pengujian performa menggunakan protokol HTTP dan HTTPS. Hasil pengujian menunjukkan bahwa *multi edge caching* secara signifikan meningkatkan kecepatan akses dan mengurangi latensi, baik pada protokol HTTP maupun HTTPS, meskipun terdapat perbedaan dalam manajemen *cache* akibat enkripsi pada HTTPS. Ini juga menunjukkan bahwa *multi edge caching* mampu meningkatkan efisiensi jaringan dan pengalaman pengguna terutama dalam lingkungan dengan permintaan konten yang tinggi.

Kata kunci : *Multi Edge Caching, Apache Traffic Server, Log Dashboard, Web Server, HTTP/HTTPS*