



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 2 (2026) pp: 13127-13138

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Simulation Prototyping Network Bonding Multi-ISP Menggunakan OpenMPTCProuter Berbasis Multipath TCP

Calvin Apriyanto*, I Putu Hariyadi, Ondi Asroni

Program Studi Ilmu Komputer, Fakultas Teknik, Universitas Bumigora, Indonesia

clvnapr@gmail.com, putu.hariyadi@universitasbumigora.ac.id, ondisembahulun@universitasbumigora.ac.id

Abstrak

Ketersediaan koneksi internet yang stabil dengan bandwidth yang memadai merupakan kebutuhan penting dalam mendukung berbagai aktivitas komunikasi data, terutama pada lingkungan yang bergantung pada akses jaringan secara berkelanjutan. Penggunaan satu Internet Service Provider (ISP) sering kali menimbulkan keterbatasan bandwidth serta risiko terjadinya gangguan layanan yang dapat mengurangi kualitas dan ketersediaan koneksi internet. Penelitian ini bertujuan untuk merancang dan mengimplementasikan network bonding multi-ISP menggunakan OpenMPTCProuter berbasis Multipath TCP (MPTCP) sebagai solusi untuk meningkatkan performa dan keandalan jaringan. Metode penelitian yang digunakan adalah Network Development Life Cycle (NDLC) yang meliputi tahapan analisis, perancangan, implementasi, simulasi, monitoring, dan pengujian sistem. Implementasi dilakukan pada lingkungan virtual menggunakan Proxmox Virtual Environment dengan dua skenario konfigurasi, yaitu 2 ISP dan 3 ISP. Evaluasi kinerja sistem dilakukan menggunakan parameter speedtest, distribusi trafik, failover, recovery, dan latency. Hasil pengujian menunjukkan bahwa konfigurasi 2 ISP menghasilkan rata-rata kecepatan download sebesar 24,24 Mbps, upload sebesar 82,98 Mbps, dan ping sebesar 48,33 ms. Sementara itu, konfigurasi 3 ISP menghasilkan rata-rata download sebesar 32,50 Mbps, upload sebesar 96,25 Mbps, dan ping sebesar 41 ms. Selain itu, mekanisme failover dan recovery mampu bekerja secara otomatis tanpa memutus koneksi pengguna, sedangkan distribusi trafik berhasil memanfaatkan seluruh jalur ISP yang aktif. Hasil penelitian membuktikan bahwa implementasi OpenMPTCProuter berbasis MPTCP efektif meningkatkan bandwidth, keandalan koneksi, serta ketersediaan layanan jaringan, dengan konfigurasi 3 ISP memberikan performa yang lebih baik dibandingkan konfigurasi 2 ISP dan layak diterapkan sebagai solusi jaringan multi-ISP.

Kata kunci: OpenMPTCProuter, Multipath TCP, Network Bonding, Multi-ISP, Failover.

1. Latar Belakang

Perkembangan teknologi informasi dan komunikasi telah meningkatkan kebutuhan terhadap layanan jaringan yang cepat, stabil, serta memiliki tingkat ketersediaan tinggi. Infrastruktur jaringan modern dituntut mampu mendukung pertumbuhan layanan berbasis cloud, virtualisasi, dan komunikasi data secara real-time sehingga diperlukan optimasi jaringan yang mampu meningkatkan performa sekaligus menjaga kontinuitas layanan jaringan. Optimalisasi jaringan menjadi faktor penting dalam menjaga efisiensi, skalabilitas, dan keandalan layanan pada lingkungan enterprise networking modern.[1]. Pada implementasinya, sebagian besar organisasi masih mengandalkan satu penyedia layanan internet atau Internet Service Provider (ISP) sebagai jalur utama akses internet. Pendekatan tersebut memiliki beberapa kelemahan seperti keterbatasan bandwidth, risiko terjadinya bottleneck trafik, serta kemungkinan terjadinya single point of failure ketika koneksi ISP mengalami gangguan. Untuk mengatasi permasalahan tersebut, banyak organisasi mulai menerapkan konsep multi-ISP guna meningkatkan ketersediaan layanan dan redundansi jaringan.[2], [3].

Salah satu teknologi yang berkembang untuk mengoptimalkan penggunaan beberapa jalur jaringan secara bersamaan adalah Multipath TCP (MPTCP). MPTCP merupakan pengembangan dari protokol TCP yang memungkinkan satu koneksi transport menggunakan beberapa jalur komunikasi secara simultan melalui pembentukan beberapa subflow pada satu sesi komunikasi. Teknologi ini mampu meningkatkan throughput, menurunkan latency, serta meningkatkan ketahanan koneksi ketika salah satu jalur mengalami gangguan [4].

MPTCP memungkinkan pemanfaatan beberapa antarmuka jaringan secara bersamaan sehingga bandwidth yang tersedia dapat digunakan secara lebih optimal dibandingkan TCP konvensional. Selain itu, MPTCP memiliki

kemampuan untuk mengalihkan trafik secara otomatis ke jalur yang masih aktif ketika terjadi kegagalan pada salah satu koneksi sehingga meningkatkan reliabilitas komunikasi data.[5]. Dimana MPTCP mampu meningkatkan throughput, menurunkan latency, dan menyediakan resiliensi yang lebih baik terhadap kegagalan jaringan melalui mekanisme penggunaan beberapa subflow secara simultan. Penelitian tersebut juga menunjukkan bahwa performa MPTCP sangat dipengaruhi oleh karakteristik jalur yang digunakan serta algoritma penjadwalan paket yang diterapkan.[6]. Para peneliti sebelumnya menunjukkan bahwa penggunaan MPTCP pada perangkat multi-homed mampu meningkatkan throughput jaringan hingga 13,6% dibandingkan metode konvensional melalui optimasi pemanfaatan beberapa jalur komunikasi secara bersamaan. Hasil tersebut menunjukkan bahwa teknologi MPTCP memiliki potensi besar dalam implementasi jaringan multi-ISP [7]. Untuk mengimplementasikan teknologi MPTCP secara praktis pada lingkungan multi-ISP, salah satu solusi yang banyak digunakan adalah OpenMPTCProuter (OMR). Salah satu implementasi praktis dari teknologi MPTCP adalah OpenMPTCProuter (OMR). OpenMPTCProuter merupakan platform open-source yang dirancang untuk menggabungkan beberapa koneksi internet menjadi satu koneksi logis menggunakan teknologi Multipath TCP [8]. Sistem ini memanfaatkan Virtual Private Server (VPS) sebagai endpoint MPTCP sehingga memungkinkan beberapa ISP digunakan secara simultan untuk meningkatkan performa jaringan dan menyediakan mekanisme failover otomatis[9].

OpenMPTCProuter merupakan platform berbasis Linux yang memanfaatkan teknologi Multipath TCP untuk menggabungkan beberapa koneksi internet menjadi satu koneksi logis melalui bantuan Virtual Private Server (VPS) [10]. Penggunaan VPS memungkinkan terbentuknya endpoint MPTCP sehingga beberapa jalur ISP dapat digunakan secara bersamaan untuk meningkatkan performa jaringan. Infrastruktur VPS sendiri telah banyak digunakan sebagai solusi virtualisasi layanan jaringan karena fleksibilitas dan kemudahan implementasinya.

2. Metode Penelitian

Penelitian ini menggunakan metode *Network Development Life Cycle* (NDLC) dimana disini untuk perancangan jaringan komputer, yang mencakup enam tahapan utama; Analysis, Design, Simulation Prototyping, Implementation, Monitoring, Management, ditahap management tidak dilakukan karena penelitian ini berfokus pada perancangan, implementasi, dan evaluasi performa sistem *network bonding* menggunakan OpenMPTCProuter pada lingkungan pengujian yang terbatas [11].



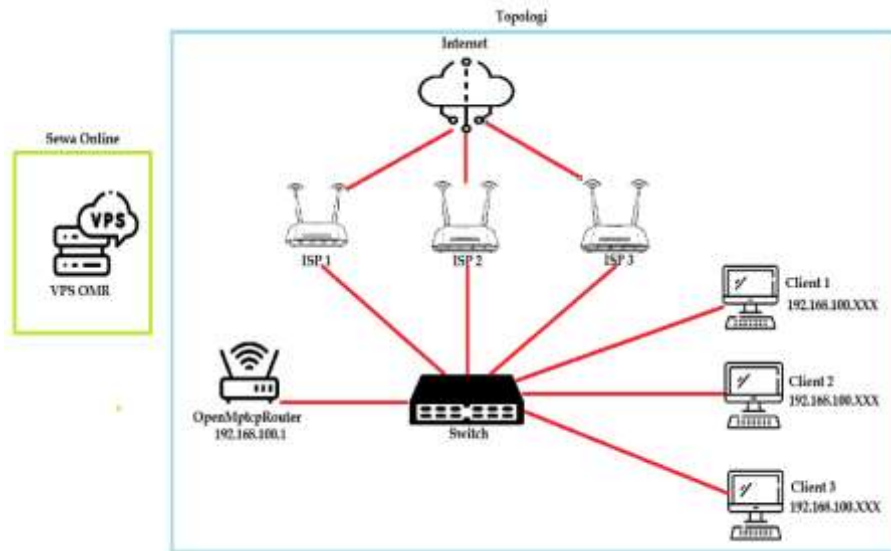
Gambar 1 Tahapan / Model NDLC

2.1. Analysis

Pada Gambar 1 di tahap analisis dilakukan pengidentifikasian kebutuhan sistem dan permasalahan dasar penelitian yang terdapat pada penggunaan satu koneksi ISP, yaitu keterbatasan bandwidth dan risiko terjadinya gangguan layanan ketika koneksi internet mengalami kegagalan, mulai dari perangkat keras, perangkat lunak, topologi jaringan, dan dilakukan studi literatur terkait implementasi Multipath TCP (MPTCP), *network bonding multi-ISP*, dan *fail over*[12].

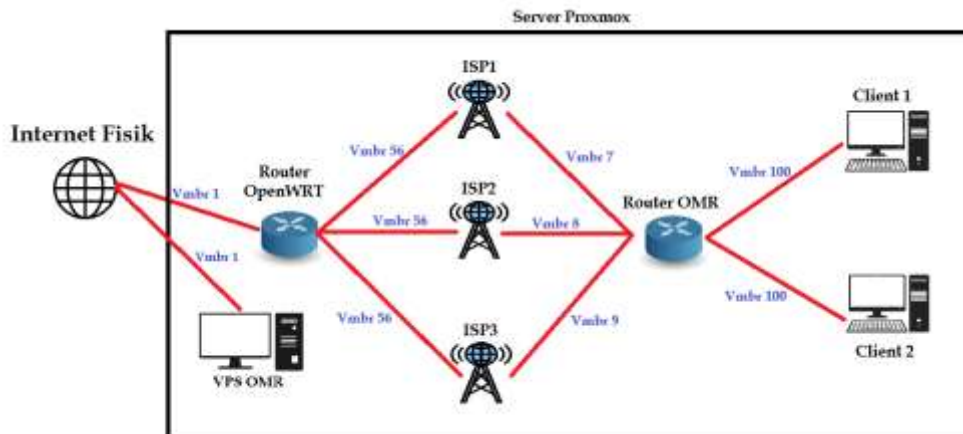
2.2. Design

Tahap design Fase perancangan mencakup perancangan arsitektur jaringan yang akan digunakan dalam penelitian ini. Perancangan tersebut meliputi topologi jaringan, skema penomoran IP, alur kerja sistem, serta persyaratan perangkat keras dan perangkat lunak. Dalam penelitian ini, OpenMPTCProuter berfungsi sebagai router utama yang menggabungkan beberapa koneksi ISP menggunakan teknologi Multipath TCP, sedangkan Virtual Private Server (VPS) digunakan sebagai titik akhir MPTCP. Desain ini dikembangkan untuk memastikan bahwa mekanisme penggabungan (bonding), distribusi lalu lintas, failover, dan pemulihan berfungsi dengan baik di lingkungan pengujian [13].



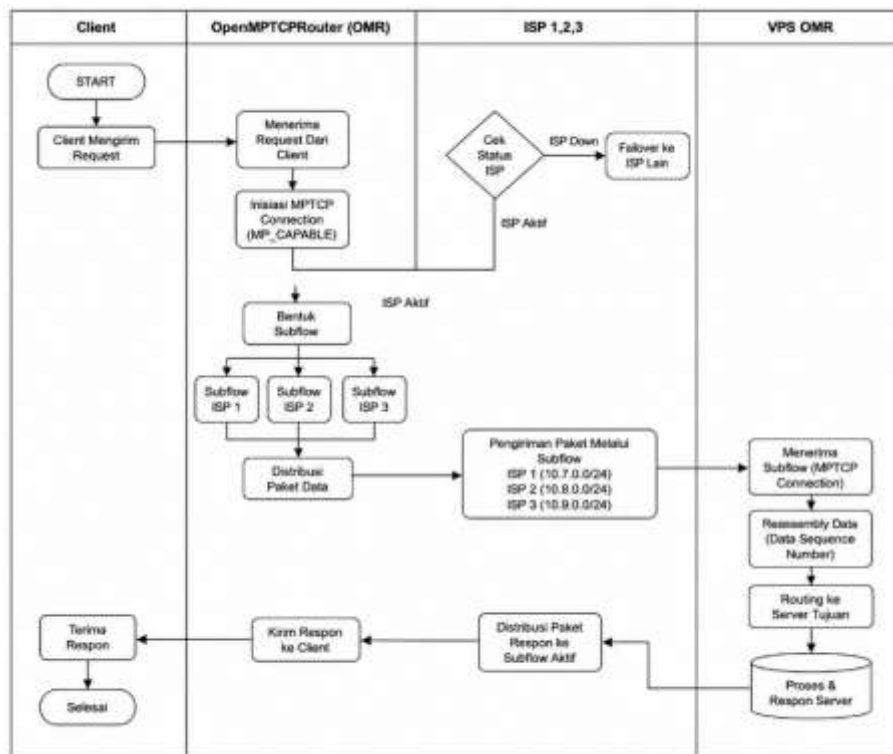
Gambar 2 Rancangan jaringan Uji Coba

Gambar 2 menunjukkan rancangan topologi jaringan yang terdiri atas OpenMPTCProuter, VPS, tiga koneksi ISP, dan client sebagai pengguna jaringan.



Gambar 3 Rancangan Topologi Dalam Virtualisasi Proxmox

Pada Gambar 3 disini menunjukkan lingkungan virtualisasi yang digunakan dalam penelitian ini. Seluruh komponen jaringan disiapkan menggunakan Proxmox Virtual Environment, sehingga simulasi dapat dilakukan secara efisien dan terkendali.[14]



Gambar 4 Rancangan Alur Sistem

Pada Gambar 4 proses komunikasi dimulai ketika perangkat klien mengirimkan permintaan untuk mengakses layanan di internet. Permintaan tersebut kemudian diteruskan ke OpenMPTCProuter, yang berfungsi sebagai gerbang utama jaringan. Setelah menerima permintaan dari klien, OMR memulai pembentukan koneksi Multipath TCP dengan server MPTCP yang berlokasi di VPS. Selama fase awal ini, terjadi proses handshake MP_CAPABLE untuk membangun koneksi MPTCP antara OMR (sebagai klien MPTCP) dan VPS (sebagai server MPTCP). Setelah koneksi MPTCP terbentuk, OMR memeriksa status koneksi setiap jalur ISP yang tersedia. Sistem kemudian menentukan jalur ISP mana yang aktif dan dapat digunakan untuk membangun koneksi. Jika salah satu jalur ISP mengalami gangguan, sistem akan secara otomatis memicu mekanisme failover dengan menggunakan jalur ISP aktif lainnya. Jika beberapa jalur ISP tersedia secara bersamaan, OMR akan membangun beberapa subaliran MPTCP melalui masing-masing jalur ISP tersebut. Setiap subaliran yang terbentuk merupakan koneksi TCP independen yang berjalan melalui jalur jaringan yang berbeda. OMR kemudian menggunakan mekanisme penjadwal MPTCP untuk mendistribusikan paket data ke seluruh subaliran yang tersedia[15].

Dengan mekanisme dari design ini, lintas data dapat dikirimkan melalui beberapa jalur ISP secara paralel, sehingga meningkatkan kinerja jaringan dan menjaga stabilitas koneksi ketika salah satu jalur mengalami gangguan. Paket data yang dikirim melalui berbagai subaliran ini kemudian diterima oleh VPS, yang bertindak sebagai titik akhir MPTCP. Di sisi VPS, sistem melakukan penyatuan kembali (reassembly) menggunakan Nomor Urutan Data (Data Sequence Number/DSN) yang terdapat dalam protokol MPTCP. Proses ini bertujuan untuk menggabungkan kembali paket data yang tiba dari berbagai subaliran menjadi satu aliran data yang utuh sebelum meneruskannya ke server tujuan di internet.

Setelah server tujuan memproses permintaan, data respons dikirim kembali ke VPS. VPS kemudian mendistribusikan paket-paket respons ini melalui beberapa subaliran aktif ke OMR. OMR selanjutnya meneruskan data respons tersebut ke perangkat klien yang mengajukan permintaan awal. Dengan mekanisme ini, sistem mampu memanfaatkan beberapa koneksi ISP secara bersamaan dan mempertahankan konektivitas jaringan melalui mekanisme failover. Selain itu, penggunaan teknologi Multipath TCP memungkinkan distribusi lalu lintas data yang efisien di seluruh jalur jaringan, sehingga meningkatkan stabilitas dan kinerja akses internet bagi pengguna.

2.3. Simulation prototyping

Tahap simulasi dilakukan dengan membangun lingkungan simulasi menggunakan platform virtualisasi Proxmox Virtual Environment, versi 7.6.4. Selama fase ini, OpenMPTCProuter, OpenWRT, sebuah Virtual Private Server (VPS), dan tiga ISP virtual yang digunakan sebagai sumber koneksi internet, diinstal dan dikonfigurasi. Setelah semua komponen berhasil dikonfigurasi, pengujian dilakukan untuk memastikan bahwa mekanisme penggabungan jaringan berfungsi sesuai dengan desain sistem. Pengujian dilakukan pada konfigurasi dengan 2 ISP dan 3 ISP untuk membandingkan kinerja jaringan yang dihasilkan. Dimana yang diuji disini ada beberapa test yang meliputi Speedtest, Mekanisme bonding, Fail over, Recovery, dan Latency.[16]

2.4. Implementation

Tahap implementasi melibatkan penerapan desain jaringan yang dibuat pada fase sebelumnya. Sistem ini diimplementasikan menggunakan platform virtualisasi Proxmox Virtual Environment (VE) versi 7.6.4 sebagai lingkungan simulasi jaringan. Selama fase ini, semua komponen yang diperlukan untuk membangun sistem penggabungan jaringan berbasis OpenMPTCProuter telah diinstal dan dikonfigurasi.

Komponen yang diimplementasikan meliputi OpenWRT sebagai gerbang internet, tiga mesin virtual yang berfungsi sebagai ISP, sebuah Virtual Private Server (VPS) sebagai titik akhir Multipath TCP, OpenMPTCProuter sebagai router utama, dan sebuah klien sebagai pengguna jaringan. Setelah semua perangkat berhasil dipasang, dilakukan konfigurasi antarmuka jaringan, penentuan alamat IP, serta konfigurasi koneksi MPTCP antara OpenMPTCProuter dan VPS[17].

Penerapan OpenMPTCProuter bertujuan untuk menciptakan beberapa subaliran MPTCP sehingga lalu lintas dapat didistribusikan ke beberapa jalur ISP secara bersamaan. Selain itu, konfigurasi ini mendukung mekanisme fail over dan pemulihan otomatis jika salah satu koneksi ISP mengalami gangguan.



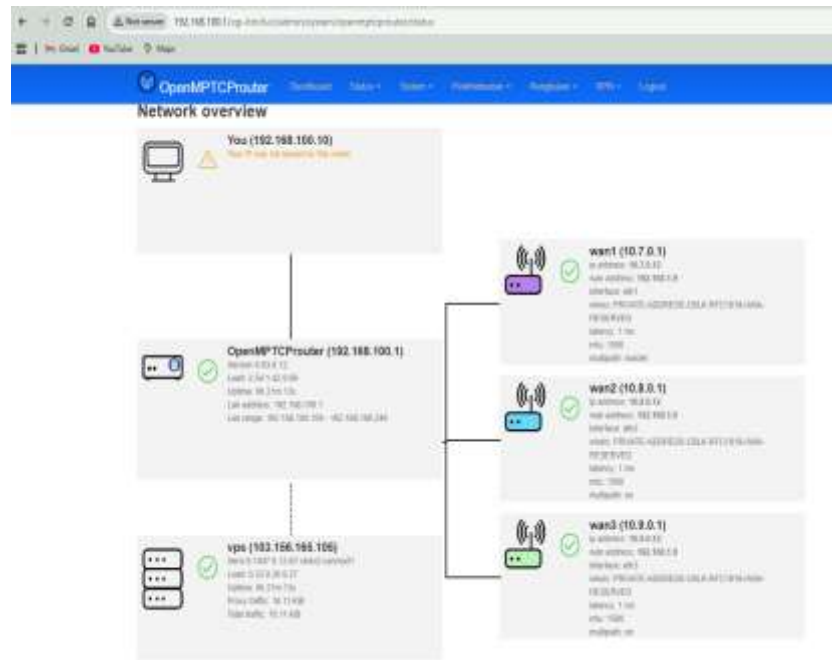
```
Debian GNU/Linux 13 vps tty1
vps login: root
Password:
Linux vps 6.12.67-x64v2-xanmod1 #0~20260123.ga077982 SMP PREEMPT_DYNAMIC Fri Jan 23 16:02:43 UTC x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
< OpenMPTCProuter VPS 0.1047 >
root@vps: # _
```

Gambar 5 Hasil Instalasi Server VPS OpenMPTCP

Pada Gambar 5 ini menunjukkan hasil instalasi VPS yang berfungsi sebagai endpoint Multipath TCP. VPS digunakan untuk menerima dan menggabungkan subflow yang dikirimkan oleh OpenMPTCProuter sebelum diteruskan menuju jaringan internet.



Gambar 6 WEB GUI OpenMPTCProuter

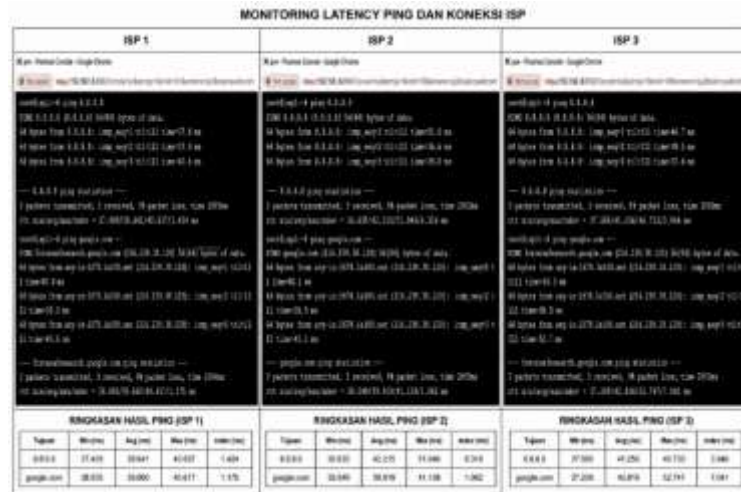
Pada Gambar 6 ini menunjukkan antarmuka Web GUI OpenMPTCProuter yang digunakan untuk memantau status koneksi ISP, pembentukan subflow, serta kondisi komunikasi antara OpenMPTCProuter dan VPS.

2.5. Monitoring

Tahap monitoring dilakukan setelah seluruh proses implementasi selesai untuk memastikan bahwa sistem penggabungan jaringan berbasis OpenMPTCProuter beroperasi sesuai dengan rancangan. Pemantauan tersebut bertujuan untuk mengamati kinerja jaringan, distribusi lalu lintas, serta kemampuan sistem dalam mempertahankan konektivitas jika terjadi gangguan pada salah satu tautan ISP.

Pada tahap ini dilakukan pengujian terhadap konfigurasi 2 ISP dan 3 ISP menggunakan beberapa parameter Quality of Service (QoS), yaitu bandwidth, latency, packet loss, failover, recovery, distribusi trafik, dan kestabilan koneksi. Pengujian dilakukan dengan memanfaatkan fitur monitoring pada OpenMPTCProuter serta beberapa tools pengujian jaringan untuk memperoleh data performa secara real-time. Hasil monitoring kemudian digunakan sebagai dasar evaluasi efektivitas implementasi network bonding menggunakan Multipath TCP (MPTCP) [18].

Selain pengujian performa, monitoring juga dilakukan untuk memastikan bahwa seluruh jalur ISP dapat membentuk subflow MPTCP dengan baik, mendistribusikan trafik secara simultan, serta menjalankan mekanisme failover dan recovery secara otomatis ketika salah satu jalur koneksi mengalami gangguan. Data yang diperoleh pada tahap ini selanjutnya dianalisis untuk mengetahui pengaruh penggunaan OpenMPTCProuter terhadap peningkatan performa dan keandalan jaringan multi-ISP.



Pada Gambar 7 menunjukkan distribusi trafik pada masing-masing jalur ISP selama proses *network bonding* berlangsung. Hasil monitoring menunjukkan bahwa OpenMPTCProuter berhasil mendistribusikan paket data melalui seluruh jalur ISP yang aktif sehingga bandwidth dapat dimanfaatkan secara lebih optimal.

Hasil penelitian Melakukan implementasi dan pengujian *network bonding* multi-ISP menggunakan OpenMPTCProuter berbasis Multipath TCP (MPTCP) dalam lingkungan virtual Proxmox. Pengujian ini dilakukan untuk mengevaluasi kemampuan sistem dalam menggabungkan beberapa koneksi Penyedia Layanan Internet (ISP) menjadi satu koneksi logis serta menjaga stabilitas jaringan ketika terjadi gangguan pada salah satu jalur komunikasi. Pengujian dilakukan dalam dua skenario konfigurasi 2-ISP dan konfigurasi 3-ISP untuk menentukan pengaruh jumlah koneksi terhadap kinerja jaringan yang dihasilkan.

3.1. Hasil Implementasi Multipath TCP

Gambar 8 Pembentukan Komunikasi antara OpenMPTCProuter dan VPS

Pada Hasil Gambar 8 pengamatan menggunakan Wireshark menunjukkan bahwa OpenMPTCProuter berhasil membentuk komunikasi Multipath TCP dengan Virtual Private Server (VPS). Pada Gambar 3.1 terlihat paket MPTCP dengan flag SYN yang dikirim melalui beberapa alamat IP yang berasal dari jalur ISP yang berbeda menuju VPS. Proses tersebut menunjukkan bahwa mekanisme pembentukan subflow MPTCP berjalan dengan baik sehingga beberapa koneksi internet dapat digunakan secara bersamaan dalam satu sesi komunikasi.

Keberhasilan pembentukan komunikasi antara OpenMPTCProuter dan VPS menjadi tahap awal yang penting dalam implementasi *network bonding* karena seluruh trafik pengguna akan diteruskan melalui subflow yang terbentuk sebelum dikirim ke jaringan internet.

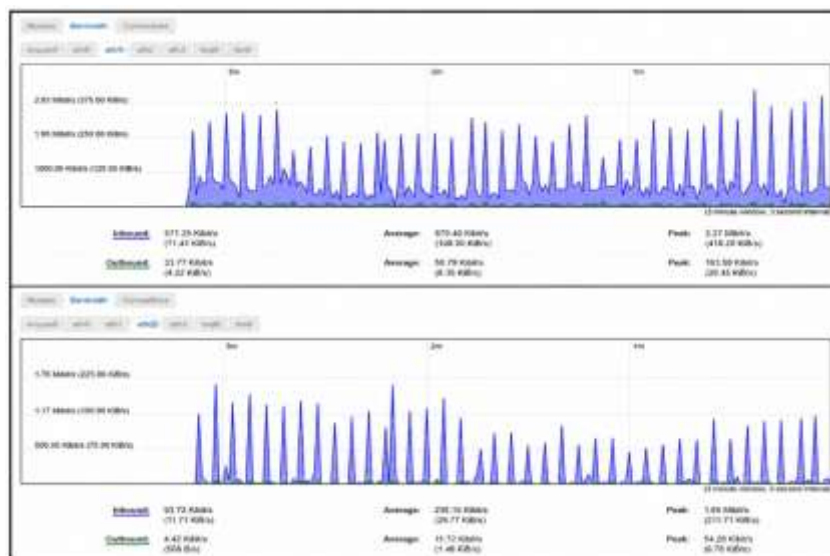
3.2. Hasil Pengujian Konfigurasi 2 ISP

Setelah koneksi MPTCP berhasil dibentuk, dilakukan pengujian menggunakan dua koneksi ISP yang terhubung ke OpenMPTCProuter. Pengujian dilakukan untuk mengetahui kemampuan sistem dalam menggabungkan bandwidth dan menjaga kelangsungan koneksi jaringan.

Tabel 3.1 Hasil Pengujian Speedtest Konfigurasi 2 ISP

Parameter	Nilai
Download	24,24 Mbps
Upload	82,98 Mbps
Ping	48,33 ms

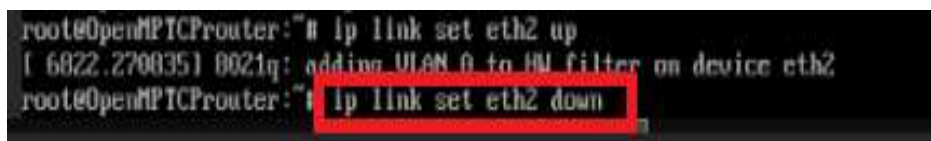
Berdasarkan hasil pada Tabel 3.1 pengujian menunjukkan bahwa konfigurasi 2 ISP menghasilkan rata-rata kecepatan download sebesar 24,24 Mbps, upload sebesar 82,98 Mbps, dan ping sebesar 48,33 ms. Nilai tersebut menunjukkan bahwa kedua jalur ISP berhasil digunakan secara simultan dalam proses komunikasi data.



Gambar 9 Monitoring Trafik Konfigurasi 2 ISP

Hasil monitoring pada Gambar 9 menunjukkan bahwa trafik berhasil didistribusikan melalui kedua jalur ISP yang aktif. Aktivitas transfer data terlihat pada masing-masing jalur koneksi selama proses pengujian berlangsung. Hal ini menunjukkan bahwa OpenMPTCProuter berhasil menjalankan mekanisme distribusi trafik melalui beberapa subflow MPTCP.

3.3 Hasil Pengujian Fail Over dan Recovery



Gambar 10 Pengujian Fail Over

Hasil pengujian pada Gambar 10 menunjukkan bahwa proses download dan streaming tetap berjalan meskipun salah satu jalur ISP mengalami gangguan. Sistem secara otomatis mengalihkan trafik ke jalur ISP yang masih aktif sehingga koneksi tidak terputus.



Gambar 11 ISP2 Kembali Dinyalakan

Ketika koneksi ISP2 di Gambar 11 yang sebelumnya mengalami gangguan kembali diaktifkan, OpenMPTCProuter berhasil melakukan recovery secara otomatis. Jalur tersebut kembali bergabung ke dalam mekanisme bonding dan digunakan kembali dalam distribusi trafik jaringan.

3.4 Hasil Pengujian Konfigurasi 3 ISP

Tahap pengujian berikutnya dilakukan dengan menambahkan satu jalur ISP sehingga OpenMPTCProuter menggunakan tiga koneksi internet secara bersamaan.



Gambar 12 Status Implementasi Network Bonding 3 ISP

Hasil implementasi Pada Gambar 12 dari ketiga koneksi ini menunjukkan bahwa OpenMPTCProuter berhasil membentuk tiga subflow aktif yang berasal dari tiga jalur ISP yang berbeda. Seluruh jalur dapat digunakan secara simultan untuk proses komunikasi data.

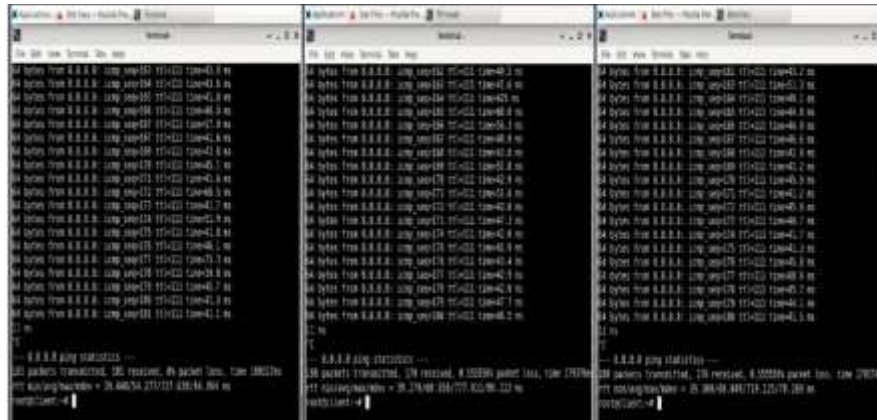
Tabel 3.2 Hasil Pengujian Speedtest Konfigurasi 3 ISP

Parameter	Nilai
Download	32,50 Mbps
Upload	96,25 Mbps
Ping	41 ms

Berdasarkan hasil Tabel 3.2 pengujian menunjukkan bahwa konfigurasi 3 ISP menghasilkan rata-rata kecepatan download sebesar 32,50 Mbps, upload sebesar 96,25 Mbps, dan ping sebesar 41 ms. Nilai tersebut lebih baik dibandingkan konfigurasi 2 ISP yang telah diuji sebelumnya.

Pada monitoring di Gambar 2.7 tentang penggabungan ISP1, ISP2, dan ISP3 menunjukkan bahwa seluruh jalur ISP aktif berpartisipasi dalam proses distribusi trafik. Trafik berhasil dikirim melalui ketiga jalur yang tersedia sehingga Gambar 13 lingga kapasitas bandwidth dapat dimanfaatkan secara lebih optimal.

3.5 Hasil Pengujian Latency



Gambar 13 Testing Latency

Tabel 3.3 Hasil Pengujian Latency

ISP	RTT Rata-rata
ISP 1	39,66 ms
ISP 2	39,91 ms
ISP 3	42,81 ms

Hasil pada Tabel 3.3 pengujian menunjukkan bahwa seluruh jalur ISP memiliki nilai RTT yang relatif stabil dengan packet loss sebesar 0%. Kondisi tersebut menunjukkan bahwa koneksi internet yang digunakan memiliki kualitas yang baik untuk mendukung implementasi Multipath TCP.

3.6 Perbandingan Hasil Pengujian Konfigurasi 2 ISP dan 3 ISP

Tabel 3.4 Perbandingan Hasil Pengujian

Parameter	2 ISP	3 ISP
Download	24,24 Mbps	32,50 Mbps
Upload	82,98 Mbps	96,25 Mbps
Ping	48,33 ms	41 ms
Bonding	Berhasil	Berhasil
Failover	Berhasil	Berhasil
Recovery	Berhasil	Berhasil

Berdasarkan hasil pengujian, kedua konfigurasi berhasil menjalankan mekanisme *network bonding* menggunakan OpenMPTCProuter. Namun, konfigurasi 3 ISP menghasilkan nilai download dan upload yang lebih tinggi serta nilai ping yang lebih rendah dibandingkan konfigurasi 2 ISP. Hasil tersebut menunjukkan bahwa penambahan jumlah jalur ISP memberikan peningkatan performa jaringan yang lebih baik.

3.7 Diskusi

Hasil penelitian menunjukkan bahwa penerapan OpenMPTCProuter berbasis Multipath TCP (MPTCP) berhasil melakukan penggabungan jaringan (*network bonding*) baik dalam konfigurasi 2-ISP maupun 3-ISP. Keberhasilan ini ditunjukkan oleh terbentuknya beberapa subaliran MPTCP antara OpenMPTCProuter dan VPS, serta kemampuan sistem untuk mendistribusikan lalu lintas ke beberapa jalur ISP secara bersamaan. Selain itu, hasil pengujian menunjukkan bahwa konfigurasi 3-ISP menghasilkan kinerja yang lebih baik daripada konfigurasi 2-ISP, dengan kecepatan unduh rata-rata 32,50 Mbps, kecepatan unggah 96,25 Mbps, dan ping 41 ms. Hasil ini menunjukkan bahwa peningkatan jumlah jalur koneksi dapat meningkatkan kapasitas bandwidth sekaligus meningkatkan kualitas komunikasi data dalam jaringan multi-ISP.

Peningkatan kinerja yang dicapai sejalan dengan prinsip kerja Multipath TCP, yang memungkinkan satu sesi komunikasi memanfaatkan beberapa jalur jaringan secara bersamaan melalui mekanisme subflow. Teknologi ini memungkinkan distribusi paket data secara paralel, sehingga kapasitas bandwidth dari beberapa ISP dapat dimanfaatkan secara lebih optimal dibandingkan saat menggunakan satu jalur komunikasi saja. Selain meningkatkan throughput, mekanisme ini juga memberikan redundansi jaringan yang lebih baik, karena koneksi tetap beroperasi meskipun salah satu jalur mengalami gangguan. Hal ini dibuktikan melalui pengujian failover dan pemulihan, yang menunjukkan bahwa baik unduhan maupun streaming tetap berlanjut meskipun salah satu ISP terputus dari jaringan.

Hasil penelitian ini sejalan dengan penelitian yang dilakukan oleh [5], yang menyatakan bahwa Multipath TCP mampu meningkatkan utilisasi bandwidth dan keandalan koneksi pada lingkungan jaringan multihomed. Penelitian [4] juga menjelaskan bahwa penggunaan beberapa jalur komunikasi secara simultan dapat meningkatkan throughput dan menyediakan resiliensi yang lebih baik terhadap kegagalan jaringan. Selain itu, penelitian Dong dkk. [4] menunjukkan bahwa optimalisasi distribusi paket pada MPTCP mampu meningkatkan performa komunikasi data melalui pemanfaatan beberapa jalur jaringan secara bersamaan.

4. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, penerapan penggabungan jaringan multi-ISP menggunakan OpenMPTCProuter berbasis Multipath TCP (MPTCP) telah berhasil diimplementasikan dan dioperasikan sesuai dengan desain sistem. OpenMPTCProuter mampu membangun beberapa subaliran MPTCP antara router dan Virtual Private Server (VPS), sehingga memungkinkan penggunaan beberapa koneksi Penyedia Layanan Internet (ISP) secara bersamaan dalam satu sesi komunikasi. Hasil pengujian menunjukkan bahwa mekanisme penggabungan jaringan, distribusi lalu lintas, failover, dan pemulihan berfungsi dengan baik tanpa menyebabkan koneksi pengguna terputus ketika salah satu tautan ISP mengalami gangguan. Pengujian performa menunjukkan bahwa konfigurasi 3 ISP menghasilkan kinerja yang lebih baik dibandingkan konfigurasi 2 ISP. Konfigurasi 2 ISP menghasilkan rata-rata kecepatan download sebesar 24,24 Mbps, upload sebesar 82,98 Mbps, dan ping sebesar 48,33 ms, sedangkan konfigurasi 3 ISP menghasilkan rata-rata download sebesar 32,50 Mbps, upload sebesar 96,25 Mbps, dan ping sebesar 41 ms. Selain itu, hasil monitoring menunjukkan bahwa seluruh jalur ISP berhasil digunakan dalam proses distribusi trafik sehingga bandwidth yang tersedia dapat dimanfaatkan secara lebih optimal. Dimana sistem ini berpotensi mengimplementasikan OpenMPTCProuter berbasis Multipath TCP terbukti mampu meningkatkan bandwidth, keandalan koneksi, dan ketersediaan layanan jaringan pada lingkungan multi-ISP. Untuk penelitian selanjutnya, disarankan melakukan pengujian dengan jumlah ISP yang lebih banyak, menerapkan algoritma penjadwalan MPTCP yang berbeda, serta melakukan evaluasi pada lingkungan jaringan nyata untuk memperoleh analisis performa yang lebih komprehensif.

Referensi

- [1] O. Afolalu, "Enterprise Networking Optimization : A Review of Challenges , Solutions , and Technological Interventions," 2025. <https://doi.org/10.3390/fi17040133>.
- [2] J. Khatib, S. Dalam, and J. Usni, "Indonesian Journal of Computer Science," vol. 11, no. 1, pp. 946–956, 2022. <https://doi.org/10.33022/ijcs.v11i3.3095>.
- [3] J. Teknologi and D. A. N. Open, "Analysis and optimization of the LAN network Using the Load Balancing Technique (Case Study : University of Muhammadiyah Palembang Campus)," vol. 8, no. 2, pp. 1165–1172, 2025. <https://doi.org/10.36378/jtos.v8i2.5166>
- [4] J. R. M. D. B. Karunarathne, T. Wijethilake, and C. Keppitiyagama, "An Evaluation of Multipath TCP with Highly Asymmetric Subflows," vol. 17, no. March, 2024. <https://doi.org/10.4038/ictcr.v17i1.7278>
- [5] L. Chao, C. Wu, T. Yoshinaga, W. Bao, and Y. Ji, "A Brief Review of Multipath TCP for Vehicular Networks," pp. 1–34, 2021. <https://doi.org/10.3390/s21082793>
- [6] Y. Li, "Improving MPTCP Performance by Detouring Network Congestion using Software-Defined Networks," no. September 2024,

- 2026, doi: 10.1145/3700523.3700551.
- [7] P. Dong *et al.*, “An Energy-Saving Scheduling Algorithm for Multipath TCP in,” pp. 1–16, 2022. <https://doi.org/10.3390/electronics11030490>
- [8] Y. Cao, R. Ji, L. Ji, M. Bao, L. Tao, and W. Yang, “Can Multipath TCP Be Robust to Cyber Attacks? A Measuring Study of MPTCP with Active Queue Management Algorithms,” vol. 2021, 2021, doi: 10.1155/2021/9963829.
- [9] A. S. Abbas, “Technical Comparison between MPTCP and TCP in Heterogeneous Networks,” vol. 16, no. 19, pp. 163–175, 2022. <https://doi.org/10.3991/ijim.v16i19.35299>
- [10] R. A. Alfarizhi, T. Ariyadi, M. Ulfa, and U. B. Dharma, “PROTOTYPE IMPLEMENTATION OF IT BUSINESS VPS AND WEB HOSTING SERVICES AS A RESEARCH LABORATORY FOR BINA DARMA UNIVERSITY IMPLEMENTASI PROTOTYPE BISNIS IT LAYANAN VPS DAN WEB HOSTING SEBAGAI LABORATORIUM”. Jurnal INOVTEK Polbeng Seri Informatika, vol. 9, no. 2, 2024.
- [11] E. Hendrawan, M. Meisel, and D. N. Sari, “ANALYSIS AND IMPLEMENTATION OF COMPUTER NETWORK SYSTEMS USING SOFTWARE DRAW . IO,” vol. 2, no. 1, pp. 9–15, 2023. DOI: 10.24042/aisj.v2i1.18227
- [12] O. Bonaventure, “Multipath TCP – documentation,” 2022. Louvain-la-Neuve, Belgium: UCLouvain, Nov. 2022. [Online]. Available: <https://www.multipath-tcp.org>
- [13] M. Firmansyah, “Design of Computer Network Infrastructure with Wired and Wireless Transmission Media Using Cisco Packet Tracer Perancangan Infrastruktur Jaringan Komputer dengan Media Transmisi Wired dan Nirkabel Menggunakan Cisco Packet Tracer,” vol. 4, no. July, pp. 1063–1071, 2024. <https://doi.org/10.57152/malcom.v4i3.1420>
- [14] I. Under, “A METHOD FOR INCREASING THE PRODUCTIVITY OF A DISTRIBUTED FIREWALL BASED ON PROXMOX,” vol. 9818, no. 3, pp. 180–188, 2025. <https://doi.org/10.30837/2522-9818.2025.3.180>
- [15] N. Al-imareen and G. Lencse, Performance Evaluation and Selection of Appropriate Congestion Control Algorithms for MPT Networks, “International Journal of Advanced Computer Science and Applications, vol. 16, no. 2, pp. 109–119, 2025.
- [16] F. Toriq and B. Santoso, “Effect of Load Balancing Bonding and Failover on Speed , Latency , Average , and Packet Loss,” vol. 8, no. 2, pp. 326–331, 2024. 10.30871/jaicv8i2.1502
- [17] T. Samosir, S. Utara, and B. G. Malau, “IMPLEMENTASI LOAD BALANCING MIKROTIK JARINGAN INTERNET DI PARDAMEAN SIBISA , AJIBATA ,” vol. 2, no. 1, pp. 20–29, 2022, doi: 10.54840/jcstech.v2i1.23.
- [18] I. Awaliyani and R. Z. A. Aziz, “Analisis Quality Of Service (QoS) Jaringan Internet Pelanggan Pada ISP Jalur data . Net Menggunakan Metode Hierarchical Token Bucket (HTB) dan Per Connection Queue (PCQ),” pp. 425–436, 2024, doi: 10.33364/algoritma/v.21-2.1792.