

## Analisis Efektivitas Failover Server Debian Menggunakan Heartbeat Cluster

Theresya Simanjuntak<sup>1\*</sup>, Jelita Astrid Gulo<sup>2</sup>, Lotar Mateus Sinaga<sup>3</sup>

<sup>1-3</sup> Program Studi Teknik Informatika, Universitas Katolik Santo Thomas Medan, Indonesia

Email: [theresyasimanjuntak0@gmail.com](mailto:theresyasimanjuntak0@gmail.com)<sup>1</sup>, [astridjelita4@gmail.com](mailto:astridjelita4@gmail.com)<sup>2</sup>, [lotarmateus88@gmail.com](mailto:lotarmateus88@gmail.com)<sup>3</sup>

\*Penulis Korespondensi: [theresyasimanjuntak0@gmail.com](mailto:theresyasimanjuntak0@gmail.com)

**Abstract** Service continuity is a critical requirement in modern network infrastructure, since even a brief interruption on a primary server can directly disrupt end-user access. This study examines the effectiveness of a failover mechanism built with Heartbeat Cluster on a Debian-based server environment. A two-node architecture was configured, consisting of Node1 as the primary (master) server and Node2 as the backup server, both monitored through Heartbeat to manage the automatic migration of a Virtual IP Address (VIP) whenever the primary node becomes unreachable. Testing was carried out by deliberately shutting down Node1 and measuring the time required for the system to transfer services to Node2 across six independent trials. The results show that Heartbeat consistently detected the failure and redirected the VIP without manual intervention, keeping the web service accessible with only a brief interruption; the average failover time recorded was 14.41 seconds. The system also performed a successful failback once Node1 was restored. These findings confirm that Heartbeat Cluster is an effective and low-cost approach for improving service availability on Debian servers, particularly for small-scale or educational network environments with limited infrastructure budgets.

**Keywords:** Debian Server; Failover; Heartbeat Cluster; High Availability; Virtual IP

**Abstrak.** Ketersediaan layanan menjadi salah satu kebutuhan utama dalam infrastruktur jaringan saat ini, karena gangguan sekecil apa pun pada server utama dapat langsung berdampak pada akses pengguna. Penelitian ini mengkaji efektivitas mekanisme failover yang dibangun menggunakan Heartbeat Cluster pada server berbasis Debian. Arsitektur dua node dikonfigurasi, yaitu Node1 sebagai server utama (master) dan Node2 sebagai server cadangan (backup), yang keduanya dipantau melalui Heartbeat untuk mengelola perpindahan Virtual IP Address (VIP) secara otomatis apabila node utama tidak lagi dapat dijangkau. Pengujian dilakukan dengan mematikan Node1 secara sengaja dan mengukur waktu yang dibutuhkan sistem untuk mengalihkan layanan ke Node2, yang diulang sebanyak enam kali percobaan independen. Hasil pengujian menunjukkan bahwa Heartbeat secara konsisten mendeteksi kegagalan dan mengalihkan VIP tanpa intervensi manual, sehingga layanan web tetap dapat diakses dengan gangguan yang relatif singkat; rata-rata waktu failover yang tercatat adalah 14,41 detik. Sistem juga berhasil melakukan failback ketika Node1 kembali aktif. Temuan ini menegaskan bahwa Heartbeat Cluster merupakan pendekatan yang efektif dan berbiaya rendah untuk meningkatkan ketersediaan layanan pada server Debian, khususnya pada lingkungan jaringan berskala kecil atau institusi pendidikan dengan keterbatasan anggaran infrastruktur.

**Kata kunci:** Debian Server; Failover; Heartbeat Cluster; High Availability; Virtual IP Address

### 1. LATAR BELAKANG

Kebutuhan akan layanan jaringan yang stabil dan dapat diakses kapan saja terus meningkat seiring semakin bergantungnya aktivitas kerja, akademik, maupun bisnis pada sistem berbasis internet. Ketika server utama yang menjalankan suatu layanan mengalami gangguan, baik karena kerusakan perangkat keras, kesalahan konfigurasi, maupun pemadaman daya, layanan yang berjalan di atasnya akan ikut terhenti sehingga pengguna kehilangan akses. Kondisi ini mendorong perlunya penerapan konsep High Availability (HA), yaitu kemampuan sistem untuk tetap beroperasi dengan tingkat downtime seminimal mungkin meskipun terjadi kegagalan pada salah satu komponennya (Riawati et al., 2022; Tareh & Zghair, 2023).

Salah satu pendekatan yang umum digunakan untuk mewujudkan HA adalah mekanisme failover clustering, yaitu proses pengalihan layanan dari server utama ke server cadangan secara otomatis ketika server utama terdeteksi tidak lagi merespons (Pribadi et al., 2020; Mukti & Nugroho, 2023). Beberapa penelitian terdahulu telah menerapkan pendekatan ini pada berbagai platform, mulai dari failover berbasis Turnkey Linux dan DRBD (Komariyah & Argyawati, 2018), pengujian performa cluster Heartbeat pada lingkungan private cloud yang berfokus pada throughput dan packet loss (Iryani et al., 2022), hingga skema failover aktif-pasif pada layanan transaksi finansial berbasis F5 (Ilham & Setiawan, 2023) dan virtualisasi server berbasis Proxmox untuk efisiensi infrastruktur (Rahman & Setiawan, 2023).

Meskipun sejumlah penelitian tersebut telah membuktikan bahwa mekanisme failover efektif meningkatkan ketersediaan layanan, sebagian besar masih dilakukan pada platform komersial, berbasis virtualisasi tingkat lanjut, atau berfokus pada aspek kinerja jaringan secara umum tanpa mengukur secara rinci konsistensi waktu perpindahan layanan pada beberapa kali percobaan berulang. Di sisi lain, kombinasi sistem operasi Debian dengan perangkat lunak Heartbeat masih relevan digunakan pada lingkungan dengan sumber daya terbatas, seperti laboratorium pendidikan maupun infrastruktur skala kecil, karena sifatnya yang ringan, open source, dan tidak memerlukan lisensi tambahan. Kesenjangan inilah yang mendasari penelitian ini, yaitu perlunya pengujian kuantitatif terhadap waktu failover dan kestabilan perpindahan Virtual IP Address (VIP) pada cluster Heartbeat berbasis Debian secara berulang, sekaligus mengaitkan hasilnya dengan parameter konfigurasi yang memengaruhi kecepatan deteksi kegagalan.

Berdasarkan uraian tersebut, penelitian ini bertujuan untuk mengimplementasikan Heartbeat Cluster pada dua node Debian Server, menguji mekanisme failover dan failback yang dihasilkan, serta menganalisis efektivitasnya berdasarkan waktu perpindahan layanan pada beberapa kali pengujian dan parameter konfigurasi yang memengaruhinya.

## **2. KAJIAN TEORITIS**

High Availability (HA) merujuk pada kemampuan suatu sistem untuk tetap beroperasi secara berkelanjutan dengan tingkat downtime yang seminimal mungkin, meskipun terjadi gangguan pada salah satu komponen penyusunnya. Konsep ini umumnya diwujudkan melalui redundansi perangkat, replikasi data, serta pemantauan sistem secara aktif, sehingga layanan tetap dapat diakses pengguna tanpa gangguan yang berarti (Riawati et al., 2022). Pada infrastruktur jaringan skala besar, penerapan HA juga banyak dikaitkan dengan manajemen

lalu lintas data agar tetap stabil ketika terjadi kegagalan pada salah satu jalur komunikasi (Taresh & Zghair, 2023).

Failover merupakan proses pengalihan beban kerja atau layanan dari server utama ke server cadangan ketika server utama mengalami kegagalan, sehingga layanan tetap dapat berjalan tanpa memerlukan intervensi administrator secara manual (Pribadi et al., 2020). Efektivitas failover umumnya diukur dari kecepatan sistem mendeteksi kegagalan serta konsistensi waktu yang dibutuhkan untuk memindahkan layanan ke node cadangan; semakin singkat dan konsisten waktu tersebut, semakin baik pula tingkat ketersediaan layanan yang dihasilkan (Mukti & Nugroho, 2023).

Heartbeat Cluster adalah salah satu perangkat lunak open source yang digunakan untuk membangun sistem cluster berbasis Linux dengan cara memantau status setiap node melalui pesan heartbeat yang dikirim secara berkala (Linux-HA Project, 2014). Apabila salah satu node tidak lagi mengirimkan sinyal dalam rentang waktu tertentu, Heartbeat akan menganggap node tersebut gagal dan secara otomatis mengalihkan resource, termasuk Virtual IP Address, ke node yang masih aktif (Komariyah & Argyawati, 2018). Kecepatan deteksi kegagalan pada Heartbeat sangat dipengaruhi oleh parameter keepalive dan deadtime yang ditetapkan pada berkas konfigurasinya, dan pengaturan kedua parameter ini turut menentukan keseimbangan antara kecepatan deteksi dan risiko perpindahan resource yang tidak diperlukan pada jaringan yang kurang stabil (Iryani et al., 2022).

Virtual IP Address (VIP) adalah alamat IP tambahan yang tidak terikat secara permanen pada satu perangkat fisik, melainkan dapat dipindahkan antar node dalam satu cluster sesuai kebutuhan (Pangestu et al., 2021). Penggunaan VIP memungkinkan klien tetap mengakses layanan melalui alamat yang sama meskipun server yang melayani permintaan tersebut telah berpindah ke node lain, sehingga tidak diperlukan perubahan konfigurasi pada sisi pengguna maupun aplikasi klien (Grevisse et al., 2022; Wicaksono & Wideasari, 2022).

Debian merupakan salah satu distribusi sistem operasi Linux yang dikenal stabil, ringan, dan banyak digunakan sebagai platform server karena dukungan repositori paket yang luas serta siklus pembaruan keamanan yang terstruktur (Debian Project, 2025). Karakteristik tersebut menjadikan Debian pilihan yang umum pada lingkungan laboratorium pendidikan maupun infrastruktur berskala kecil, termasuk dalam simulasi jaringan berbasis virtualisasi seperti VirtualBox (Santoso et al., 2022; Sugiyanto et al., 2021), sebelum diterapkan pada lingkungan produksi yang lebih kompleks (Nemeth et al., 2017).

### 3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimental, yaitu membangun secara langsung sebuah sistem cluster berbasis Heartbeat pada dua unit Debian Server dan menguji perilakunya melalui simulasi kegagalan. Seluruh node dijalankan sebagai mesin virtual menggunakan VirtualBox pada satu perangkat host, dengan pertimbangan bahwa virtualisasi memudahkan replikasi lingkungan pengujian secara konsisten tanpa memerlukan perangkat keras fisik tambahan (Sugiyanto et al., 2021).

Perangkat keras yang digunakan meliputi satu unit laptop dengan RAM minimal 8 GB dan ruang penyimpanan tersedia minimal 50 GB, yang menjadi host bagi kedua mesin virtual. Perangkat lunak yang digunakan terdiri atas VirtualBox sebagai platform virtualisasi, dua instansi Debian 12 (Node1 dan Node2), Heartbeat sebagai perangkat lunak cluster, serta Apache2 sebagai layanan web yang diuji ketersediaannya.

Topologi sistem dirancang dengan Node1 (192.168.56.11) berperan sebagai server utama (master) dan Node2 (192.168.56.12) sebagai server cadangan (backup). Kedua node berbagi satu Virtual IP Address (192.168.56.100) yang dikelola oleh Heartbeat dan digunakan klien untuk mengakses layanan Apache2, sehingga perpindahan layanan antar node tidak memerlukan perubahan alamat pada sisi klien.

Tahapan penelitian dilaksanakan secara berurutan, meliputi: (1) perancangan topologi jaringan antar node; (2) konfigurasi alamat jaringan pada masing-masing node; (3) instalasi paket Heartbeat pada Node1 dan Node2; (4) konfigurasi berkas authkeys sebagai mekanisme autentikasi antar node; (5) konfigurasi berkas ha.cf untuk menentukan parameter cluster seperti keepalive dan deadtime; (6) konfigurasi berkas haresources untuk menentukan node utama beserta resource yang dikelola; (7) pengujian mekanisme failover dengan mematikan Node1 secara sengaja; dan (8) analisis hasil pengujian berdasarkan waktu perpindahan VIP yang tercatat pada enam kali percobaan berulang.

Data utama yang dikumpulkan adalah waktu yang dibutuhkan sistem untuk memindahkan VIP dari Node1 ke Node2 setelah Node1 dimatikan, yang diukur sejak perintah poweroff dijalankan hingga VIP terdeteksi aktif pada Node2 melalui perintah ip a. Data dari enam percobaan tersebut kemudian direkapitulasi dalam bentuk tabel dan grafik, lalu dianalisis secara deskriptif untuk melihat rata-rata, sebaran, serta faktor konfigurasi yang memengaruhi variasi waktu failover yang terjadi.

#### 4. HASIL DAN PEMBAHASAN

##### **Pengujian Konektivitas dan Konfigurasi Dasar Cluster**

Sebelum layanan cluster dijalankan, konektivitas antar node terlebih dahulu diuji menggunakan perintah ping untuk memastikan Node1 dan Node2 dapat saling berkomunikasi melalui jaringan yang telah dikonfigurasi. Hasil pengujian menunjukkan bahwa kedua node saling terhubung tanpa kehilangan paket, sehingga proses instalasi dan konfigurasi Heartbeat dapat dilanjutkan. Berkas authkeys dikonfigurasi sebagai mekanisme autentikasi agar hanya node yang telah terdaftar yang dapat bergabung dalam cluster, berkas ha.cf digunakan untuk menetapkan parameter keepalive, deadtime, serta daftar node anggota cluster, dan berkas haresources digunakan untuk menentukan node utama beserta Virtual IP Address dan layanan yang dikelola oleh Heartbeat. Setelah seluruh konfigurasi diterapkan, layanan Heartbeat pada kedua node berhasil berjalan dengan status active (running), yang menandakan bahwa cluster telah terbentuk dan siap menjalankan mekanisme failover.

##### **Mekanisme Kerja Failover**

##### ***Kondisi Normal: Virtual IP pada Node1***

Pada kondisi normal, Virtual IP Address 192.168.56.100 berada pada Node1 selaku server utama, dan layanan web yang dijalankan melalui Apache2 dapat diakses klien melalui alamat VIP tersebut. Kondisi ini menunjukkan bahwa Node1 secara aktif melayani seluruh permintaan yang masuk selama tidak terjadi gangguan.

##### ***Simulasi Kegagalan dan Perpindahan VIP ke Node2***

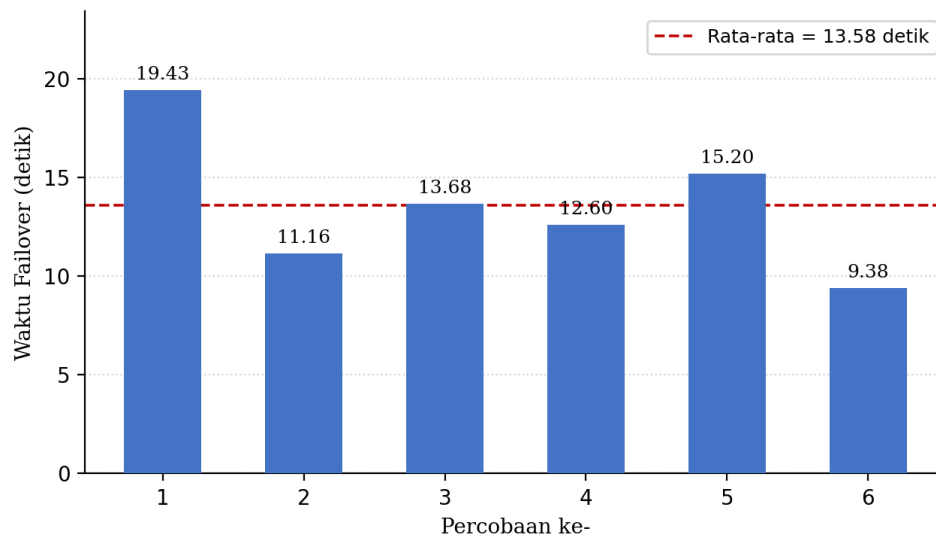
Simulasi kegagalan dilakukan dengan mematikan Node1 menggunakan perintah poweroff untuk merepresentasikan kondisi server utama yang tidak lagi dapat diakses. Heartbeat pada Node2 mendeteksi hilangnya sinyal heartbeat dari Node1 setelah rentang waktu deadtime terlampaui, kemudian secara otomatis mengambil alih Virtual IP Address dan mengaktifkan layanan Apache2 pada Node2. Pengujian akses web setelah proses ini menunjukkan bahwa layanan tetap dapat dijangkau klien melalui alamat VIP yang sama tanpa perubahan konfigurasi apa pun pada sisi klien, yang mengonfirmasi bahwa mekanisme failover berjalan sebagaimana dirancang.

##### **Hasil Pengujian Waktu Failover**

Pengujian waktu failover dilakukan sebanyak enam kali percobaan independen dengan cara yang sama, yaitu mematikan Node1 dan mencatat waktu hingga Virtual IP Address terdeteksi aktif pada Node2. Rekapitulasi hasil pengujian disajikan pada Tabel 1.

**Tabel 1.** Hasil Pengukuran Waktu Failover

| Percobaan ke- | Waktu Failover (detik) |
|---------------|------------------------|
| 1             | 19,43                  |
| 2             | 11,16                  |
| 3             | 13,68                  |
| 4             | 12,60                  |
| 5             | 15,20                  |
| 6             | 09,38                  |

**Gambar 1.** Grafik Hasil Pengujian Waktu Failover

Data pada Tabel 1 dan Gambar 1 menunjukkan bahwa waktu failover pada enam kali percobaan bervariasi antara 9,38 detik hingga 19,43 detik, dengan rata-rata 14,41 detik. Variasi ini tergolong wajar mengingat proses deteksi kegagalan pada Heartbeat bergantung pada siklus pengiriman sinyal (keepalive) dan ambang batas waktu tunggu (deadtime) yang telah ditetapkan sebelumnya, sehingga selisih waktu antara satu percobaan dengan percobaan lainnya dapat dipengaruhi oleh posisi node dalam siklus pengiriman sinyal pada saat kegagalan disimulasikan.

### Analisis dan Pembahasan Hasil

Secara keseluruhan, mekanisme failover pada penelitian ini berjalan sesuai dengan konfigurasi yang diterapkan. Ketika Node1 dimatikan, Heartbeat pada Node2 berhasil mendeteksi kegagalan komunikasi dan mengalihkan Virtual IP Address secara otomatis, sehingga layanan Apache2 tetap dapat diakses melalui alamat yang sama tanpa memerlukan intervensi administrator. Temuan ini sejalan dengan penelitian Pribadi et al. (2020) dan Komariyah & Argyawati (2018) yang juga melaporkan bahwa mekanisme failover berbasis

Heartbeat mampu menjaga ketersediaan layanan web ketika server utama mengalami gangguan.

Selain proses failover, sistem pada penelitian ini juga berhasil melakukan failback, yaitu mengembalikan Virtual IP Address ke Node1 setelah node tersebut kembali aktif. Kemampuan ini penting karena memastikan beban layanan kembali terdistribusi sesuai peran awal masing-masing node, sejalan dengan prinsip pemulihan otomatis yang juga ditekankan pada penelitian mengenai skema failover aktif-pasif oleh Ilham & Setiawan (2023).

Variasi waktu failover yang tercatat pada penelitian ini turut menegaskan bahwa parameter *keepalive* dan *deadtime* memiliki peran penting dalam menentukan kecepatan deteksi kegagalan. Semakin kecil nilai *deadtime* yang ditetapkan, semakin cepat pula sistem mendeteksi kegagalan node, namun risiko terjadinya perpindahan resource yang tidak diperlukan akan meningkat apabila kondisi jaringan tidak stabil, sebagaimana juga diamati pada pengujian performa cluster Heartbeat oleh Iryani et al. (2022) dan pada penelitian mengenai load balancing berbasis klaster oleh Mukti & Nugroho (2023). Dengan demikian, konfigurasi kedua parameter tersebut perlu disesuaikan dengan karakteristik jaringan tempat cluster diterapkan agar diperoleh keseimbangan antara kecepatan respons dan kestabilan sistem, sejalan pula dengan prinsip manajemen lalu lintas pada infrastruktur high availability yang dikemukakan oleh Taresh & Zghair (2023).

Dibandingkan dengan penelitian-penelitian sebelumnya yang umumnya menerapkan failover pada platform virtualisasi tingkat lanjut atau perangkat komersial, hasil penelitian ini menunjukkan bahwa kombinasi Debian Server dan Heartbeat tetap mampu memberikan mekanisme high availability yang efektif dengan sumber daya dan biaya yang jauh lebih rendah, sehingga relevan diterapkan pada lingkungan laboratorium pendidikan maupun infrastruktur jaringan berskala kecil.

## 5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengimplementasikan Heartbeat Cluster pada dua node Debian Server untuk membangun mekanisme failover berbasis Virtual IP Address. Hasil pengujian pada enam kali percobaan menunjukkan bahwa sistem secara konsisten mampu mendeteksi kegagalan pada Node1 dan mengalihkan layanan ke Node2 dengan rata-rata waktu failover 14,41 detik, tanpa memerlukan perubahan konfigurasi pada sisi klien. Sistem juga terbukti mampu melakukan failback ketika Node1 kembali aktif, sehingga peran server utama dapat dipulihkan secara otomatis. Temuan ini menunjukkan bahwa Heartbeat Cluster efektif

digunakan sebagai solusi high availability berbiaya rendah pada lingkungan Debian Server, khususnya untuk kebutuhan skala kecil hingga menengah.

Penelitian ini masih memiliki keterbatasan, antara lain jumlah percobaan yang relatif sedikit, pengujian yang hanya dilakukan pada satu jenis layanan yaitu Apache2, serta lingkungan pengujian yang sepenuhnya berbasis virtualisasi pada satu perangkat host sehingga belum sepenuhnya merepresentasikan kondisi jaringan produksi dengan variasi latensi dan beban yang lebih kompleks. Oleh karena itu, penelitian selanjutnya disarankan untuk menambah jumlah node guna meningkatkan redundansi sistem, menguji mekanisme failover pada layanan lain seperti basis data atau layanan berbasis API, mengintegrasikan sistem pemantauan jaringan secara real-time, serta melakukan pengujian pada kondisi jaringan yang lebih bervariasi untuk mengevaluasi lebih jauh pengaruh parameter keepalive dan deadtime terhadap kestabilan cluster.

### **Ucapan Terima Kasih**

Penulis mengucapkan terima kasih kepada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas Medan atas dukungan fasilitas laboratorium yang digunakan selama proses penelitian ini. Artikel ini merupakan pengembangan lebih lanjut dari laporan praktikum administrasi dan keamanan jaringan yang telah dilaksanakan oleh penulis.

### **DAFTAR REFERENSI**

- Andreatos, A. (2023). Educational scenario for teaching cyber security using low-cost equipment and open source software. *Proceedings of the European Conference on Cyber Warfare and Security*, 22(1), 44–53.
- Dame, A. H., & Zailani, A. U. (2023). Implementasi webserver berbasis Docker dan Linux. *Jurnal Ilmu Komputer dan Pendidikan*, 1(5), 1–8.
- Debian Project. (2025). Debian administrator's handbook. <https://www.debian.org/doc/>
- Fikri, M., & Rifqi, M. (2023). Implementasi VPN antar cabang menggunakan teknologi SD-WAN dengan metode load balance (Studi kasus: PT. Mitra Solusi Infokom). *Jurnal Teknologi Informasi dan Ilmu Komputer*, 10(1), 105–113. <https://doi.org/10.25126/jtiik.2023105236>
- Grevisse, D. Y. R., Emmanuel, K. K., Cedrick, M. M., Freddy, K., & Raymond-Albert, L. A. T. (2022). Implementation of a secure remote administration system in a local network: Simulated application to university institutions. *International Journal of Computer Applications*, 184(20), 1–7.
- Ilham, B., & Setiawan, Y. (2023). Implementation of high availability message ISO 8583 using F5 active-passive failover method. *International Journal of Engineering Trends and Technology*, 71(4), 264–273. <https://doi.org/10.14445/22315381/IJETT-V71I4P223>

- Iryani, N., Ayatri, K. D., & Wahyuningrum, R. D. (2022). Analisis performansi high availability cluster server menggunakan heartbeat pada private cloud. *JITEL (Jurnal Ilmiah Telekomunikasi, Elektronika, dan Listrik Tenaga)*, 2(2), 55–62.
- Komariyah, F., & Argyawati, H. (2018). Implementasi server cluster high availability pada web server dengan sistem operasi Turnkey Linux menggunakan Heartbeat. *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, 4(2), 78–88.
- Linux-HA Project. (2014). Heartbeat: Subsystem for high-availability Linux. <http://linux-ha.org/>
- Mukti, F. S., & Nugroho, F. E. (2023). Upaya pencapaian status high availability server menggunakan metode load balancing berbasis klaster pada database server PT. XYZ. *Jurnal Teknik Informatika dan Sistem Informasi*, 10(1), 757–767.
- Nemeth, E., Snyder, G., Hein, T. R., & Whaley, B. (2017). *Unix and Linux system administration handbook* (5th ed.). Addison-Wesley.
- Novianto, D., Japriadi, Y. S., Pangkalpinang, A. L., Sudirman, J., Baru, K. S., Gabek, K., & Pinang, K. P. (2021). Comparative analysis of performance between ECMP and NTH methods in implementation of Mikrotik-based dual link load balancing techniques. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, 10(3), 210–217.
- Pangestu, R. N., Yanti, R., & Harahap, H. (2021). Implementasi keamanan jaringan berbasis VPN dan anti-DDoS dalam melindungi server Linux dari serangan Hammer. *Jurnal Ilmu Komputer dan Sistem Komputer Terapan*, 3(1), 1–9. <https://doi.org/10.35447/jikstra.v3i1.357>
- Rahman, A., & Setiawan, B. (2023). Implementasi virtualisasi server menggunakan Proxmox VE untuk efisiensi infrastruktur jaringan. *Jurnal Teknologi Informasi dan Komputer*, 7(2), 45–52.
- Riawati, A. D., Irfan, M., Khaeruddin, & Faruq, A. (2022). High availability dynamic sharding database server dengan metode fail over dan clustering. *Jurnal Manajemen Informatika & Sistem Informasi*, 5(1), 1–10.
- Santoso, N. A., Zakaria, & Kurniawan, R. D. (2022). Analisis jaringan komputer menggunakan teknologi virtualisasi. *Jurnal Minfo Polgan*, 11(2), 52–58. <https://doi.org/10.33395/jmp.v11i2.11652>
- Sugiyanto, H., Nugraha, A. C., & Prasetyo, D. (2021). Perbandingan performa perangkat lunak virtualisasi Oracle VirtualBox dan VMware Workstation. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 5(2), 329–335.
- Tanenbaum, A. S., & Wetherall, D. J. (2021). *Computer networks* (6th ed.). Pearson.
- Taresh, A. A. R., & Zghair, N. A. K. (2023). Redesign of the communications network based on high availability of traffic management technologies to improve the communication. *Measurement: Sensors*, 27, Article 100776. <https://doi.org/10.1016/j.measen.2023.100776>
- Wibowo, A. R., & Dewi, M. S. A. (2021). Pengujian performa database MySQL dan PostgreSQL dalam lingkungan virtualisasi. *Jurnal Teknologi Informasi dan Komputer*, 7(2), 77–84.
- Wicaksono, D., & Widiyari, I. R. (2022). Sistem keamanan jaringan menggunakan firewall dengan metode port blocking dan firewall filtering. *Jurnal Teknik Informatika dan Sistem Informasi*, 9(2), 1380–1392.