

Implementasi Web Server Casaos Menggunakan Docker Dan Cloudflare Tunnels Pada Linux Ubuntu

Rokiman^{1*}, Aziz Azindani², Sendie Yulianto Margen³

¹⁻³ Program Studi Teknik Informatika, Politeknik Baja Tegal

Alamat: Jl. Raya Barat Dukuhwaru, Jatibarang-Slawi Km 7, Kab Tegal

Email: rokiman1801@gmail.com*

Abstract. *Web servers often face high operational costs and reliance on expensive third-party services, which are inefficient in terms of cost and security. This study addresses these issues by developing a CasaOS- based web server using Docker and Cloudflare Tunnels technology on Linux Ubuntu. The Research and Development (R&D) method was used, involving design, implementation, and testing stages. The implementation process included installing Linux Ubuntu, configuring static IPs, installing CasaOS, managing applications with Docker, and secure access through Cloudflare Tunnels. Testing was conducted on native websites and WordPress to evaluate ease of use and system performance. The results show that this web server provides high efficiency, better security, and ease of management without additional costs, effectively addressing high operational costs and reliance on third-party services.*

Keywords: *web server, CasaOS, Docker, Cloudflare Tunnels, Linux Ubuntu.*

Abstrak. *Web server sering menghadapi biaya operasional yang tinggi dan ketergantungan pada layanan pihak ketiga yang mahal, yang tidak efisien dari segi biaya dan keamanan. Penelitian ini mengatasi masalah tersebut dengan mengembangkan web server berbasis CasaOS menggunakan teknologi Docker dan Cloudflare Tunnels pada Linux Ubuntu. Metode Research and Development (R&D) digunakan dengan tahapan perancangan, implementasi, dan pengujian. Proses implementasi mencakup instalasi Linux Ubuntu, konfigurasi IP statis, instalasi CasaOS, manajemen aplikasi dengan Docker, dan akses aman melalui Cloudflare Tunnels. Pengujian dilakukan pada website native dan WordPress untuk mengevaluasi kemudahan penggunaan dan performa sistem. Hasilnya menunjukkan bahwa web server ini memberikan efisiensi tinggi, keamanan yang lebih baik, dan kemudahan pengelolaan tanpa biaya tambahan, secara efektif mengatasi biaya operasional yang tinggi dan ketergantungan pada layanan pihak ketiga.*

Kata Kunci: *web server, CasaOS, Docker, Cloudflare Tunnels, Linux Ubuntu.*

PENDAHULUAN

Membangun *web server* sendiri memungkinkan perusahaan menghemat biaya operasional serta memiliki kendali penuh atas data dan sistem yang digunakan. Hal ini berbeda dengan ketergantungan pada layanan pihak ketiga yang mahal dan berisiko. Penelitian sebelumnya (Ihsan et al., 2023) menunjukkan berbagai tantangan dalam implementasi *web server*. Penelitian menunjukkan ketergantungan pada layanan *cloud* dan biaya tinggi. (Susilo et al., 2019)

mengembangkan *web server* untuk pembelajaran di SMK, namun menghadapi kompleksitas implementasi dan kebutuhan *hardware* yang tinggi (Mangayarkarasi et al., 2021) menggunakan *Docker Swarm* untuk *load balancing*, dengan kelebihan dalam skalabilitas namun kompleks dalam pengelolaan. (Ariadi et al., 2020) menunjukkan efisiensi *Docker Container*, namun kurang dalam integrasi pengelolaan *file*.

Penelitian ini mengusulkan solusi *web server* mandiri menggunakan *CasaOS*, *Docker*, dan *Cloudflare*, yang menawarkan antarmuka

pengguna yang ramah, isolasi aplikasi, efisiensi penggunaan sumber daya, serta peningkatan keamanan dan kinerja. Permasalahan yang diangkat dalam penelitian ini adalah bagaimana merancang *web server* menggunakan *CasaOS*, *Docker*, dan *Cloudflare* pada sistem operasi *Linux Ubuntu*, serta bagaimana implementasinya dapat dilakukan dengan mudah dan ramah pengguna.

Penelitian ini bertujuan merancang dan mengimplementasikan *web server* berbasis *CasaOS* dengan *Docker* dan *Cloudflare* pada *Linux Ubuntu* untuk solusi yang lebih ekonomis, efisien, dan aman, serta memberikan kendali penuh atas data dan sistem. Originalitas penelitian terletak pada kombinasi *CasaOS*, *Docker*, dan *Cloudflare* untuk menciptakan *web server* mandiri yang inovatif, mengatasi masalah biaya dan ketergantungan pada pihak ketiga.

LANDASAN TEORI

Implementasi

Implementasi merupakan sebuah penerapan pengetahuan, metode, atau teknologi baru untuk mencapai tujuan tertentu. Sistem informasi manajemen mengorganisasi dan menyajikan data untuk mendukung keputusan dan operasional organisasi. Tujuan dari implementasi ini adalah meningkatkan efektivitas, efisiensi, dan kualitas hasil. (Daheri, 2023).

Web Server

Server web adalah perangkat lunak yang menerima permintaan HTTP dari pengguna melalui *browser* dan mengirimkan kembali halaman *web* yang berisi teks, video, gambar, animasi, dan data audio (Saraswati et al., 2019).

Apache

Apache adalah salah satu aplikasi *web server open source* yang paling banyak digunakan dan dapat dijalankan di banyak

sistem operasi seperti *Windows*, *Linux*, dan *Mac*. *Web server* ini berguna untuk melayani dan memfungsikan situs web dengan mengelola dan mengirimkan konten *web* kepada pengguna melalui internet (Kesuma Astuti et al., 2022).

PHP

PHP adalah bahasa pemrograman yang digunakan secara luas untuk pembuatan dan pengembangan *web*. PHP dapat disertakan dalam dokumen HTML dan bekerja di sisi *server* untuk menghasilkan halaman *web* yang dinamis (Kesuma Astuti et al., 2022). PHP, singkatan dari "PHP: *Hypertext Preprocessor*", adalah bahasa skrip yang dirancang khusus untuk pengembangan web. PHP bersifat *open source*, memungkinkan pengguna untuk mengembangkan kode PHP sesuai kebutuhan mereka. Bahasa ini pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1994 dan sejak itu telah berkembang menjadi salah satu bahasa pemrograman yang paling populer di dunia untuk pengembangan web.

MySQL

MySQL populer di kalangan pengembang web karena pencarian data yang cepat dan integrasi yang baik dengan PHP, menjadikannya pilihan utama untuk proyek web *modern*. Performa MySQL dapat ditingkatkan melalui optimasi, seperti penggunaan indeks dan optimasi *query*, yang sangat efektif untuk *volume* data besar (Györödi et al., 2021; Maesaroh et al., 2022).

CasaOS

CasaOS adalah sistem operasi Linux untuk server rumah, dirancang oleh IceWhale Technology pada 2021. Antarmuka ramah pengguna dan manajemen layanan yang

mudah memungkinkan pembangunan server rumah yang hemat energi dan biaya, namun tetap handal dan efisien. (Tech, 2024).

Docker

Docker menyederhanakan pembuatan, penyebaran, dan pelaksanaan aplikasi menggunakan *container*. *Container* menjalankan aplikasi dalam lingkungan virtual yang terisolasi, memastikan kompatibilitas di berbagai *platform* (Docker, 2020). *Docker* menyediakan alat seperti *Docker Engine*, *Docker Hub*, dan *Docker Compose* untuk mengelola *container* (Miell et al., 2019).

Cloudflare Tunnels dan Zero Trust

Cloudflare Tunnels merupakan solusi efektif untuk meningkatkan keamanan dan kinerja aplikasi web melalui penggunaan protokol *tunneling* dan enkripsi. Adopsi teknologi ini memungkinkan perusahaan melindungi data mereka dari ancaman eksternal tanpa mengorbankan kemudahan penggunaan dan kinerja aplikasi (Ibrahim, 2023).

Zero Trust merupakan model keamanan yang mengasumsikan bahwa tidak ada elemen jaringan, baik internal maupun eksternal, yang sepenuhnya dipercaya. Pendekatan ini bertujuan untuk mengurangi risiko keamanan dengan memvalidasi setiap permintaan akses berdasarkan beberapa faktor autentikasi dan otorisasi sebelum memberikan akses ke sumber daya (Rose et al., 2022).

Linux Ubuntu

Ubuntu merupakan distribusi Linux yang dikembangkan oleh Canonical Ltd. yang dikenal dengan stabilitas, keamanan, dan kemudahan penggunaannya. Versi terbaru, Ubuntu 22.04 LTS (*Long Term Support*), memberikan peningkatan signifikan baik dari segi performa maupun fitur keamanan yang lebih canggih. Ubuntu 22.04 telah terbukti memiliki kinerja yang unggul dalam hal pengelolaan sumber daya sistem. Evaluasi kinerja menunjukkan bahwa distribusi ini

mampu mengelola penggunaan CPU dan memori dengan efisien, yang berkontribusi pada stabilitas dan kecepatan sistem operasi (Sinha et al., 2023). Penggunaan kernel *Linux* versi terbaru juga mendukung perangkat keras modern dengan optimal, meningkatkan performa keseluruhan sistem (Boras et al., 2020).

METODE PENELITIAN

Penelitian ini merupakan *Research and Development* (R&D) yang melibatkan tahapan sistematis untuk menghasilkan dan mengevaluasi produk atau solusi baru. Metode ini mencakup Tahap Perancangan, Implementasi, dan Pengujian, yang didasarkan pada penelitian mendalam dan analisis cermat (Kostrzewa-Nowak et al., 2023).

Perencanaan

Dilakukan analisis kebutuhan yang mendalam untuk memastikan solusi yang diusulkan dapat memenuhi semua persyaratan yang relevan. Analisis ini mencakup kebutuhan teknis, perangkat keras, dan perangkat lunak dasar yang diperlukan

Kebutuhan Teknis dan Teknologi

Untuk mengimplementasikan *web server* berbasis *CasaOS* dengan *Docker* dan *Cloudflare Tunnels* pada *Linux Ubuntu*, diperlukan analisis kebutuhan mendalam untuk memastikan solusi memenuhi semua persyaratan relevan. Analisis ini mencakup kebutuhan teknis seperti pada Tabel 3.1 dan teknologi seperti ditunjukkan pada Tabel 3.2 yang akan dibahas secara terperinci untuk mencapai tujuan penelitian ini.

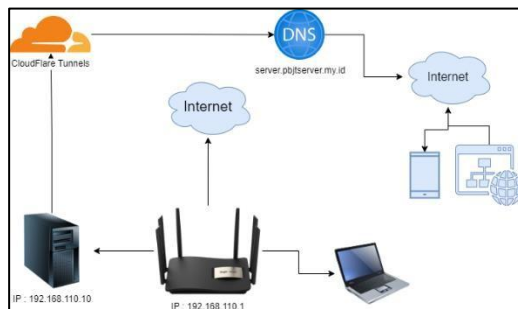
Tabel 3. 1 Kebutuhan Teknis

CPU	Intel i3-3240 4core 3400GHz
RAM	8 GB
SSD	128 GB
Router	Ruijie
Kabel UTP	UTP Cat8

Tabel 3. 2 Kebutuhan Teknologi

Sistem Operasi	Linux Ubuntu
Software	CasaOS
	Docker
	MySQL
	phpMyAdmin

Topologi Jaringan



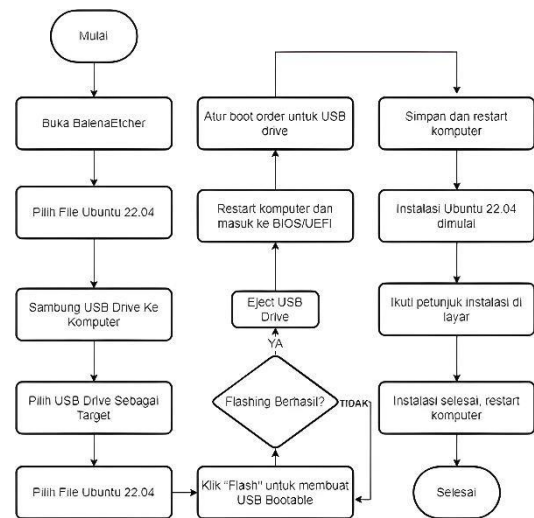
Gambar 3. 1 Topologi Jaringan

Implementasi

Implementasi mencakup perakitan perangkat keras, instalasi sistem operasi, dan konfigurasi perangkat lunak yang diperlukan. Tahapan ini meliputi Konfigurasi *CasaOS*, *Docker*, dan *Cloudflare Tunnels* pada sistem operasi *Linux Ubuntu*.

A. Instalasi Linux Ubuntu

Instalasi dilakukan dengan menggunakan metode bootable. Seperti pada gambar 3.2 menggambarkan instalasi Linux Ubuntu secara sistematis. Sebelum memulai proses instalasi, pastikan perangkat keras sudah terhubung dengan benar dan media bootable berfungsi dengan baik. Langkah-langkah instalasi dimulai dengan memasukkan media bootable, memilih opsi boot dari BIOS, dan mengikuti instruksi yang muncul di layar untuk menginstal Linux Ubuntu hingga selesai. Setelah sistem operasi berhasil diinstal, langkah selanjutnya adalah melakukan pembaruan sistem dan menginstal paket-paket dasar yang diperlukan. Selain itu, konfigurasi IP statis juga dilakukan untuk memastikan server memiliki alamat IP yang tetap.



Gambar 3. 2 Instalasi Linux Ubuntu

B. Konfigurasi Docker

Docker Compose web native dan wordpress

```
web2:
  build:
    context: ./php
  dockerfile: Dockerfile
  container_name: native
  restart: always
  ports:
    - "8400:80"
  volumes:
    - ./web/native:/var/www/html
  depends_on:
    - db
  networks:
    - network

poltek:
  image: wordpress:latest
  container_name: poltek
  restart: always
  ports:
    - "8100:80"
  depends_on:
    - db
  volumes:
    - ./wordpress/poltek:/var/www/html
  networks:
    - network
```

C. Konfigurasi Cloudflare Tunnels

```
3 poltek.pbjtservers.my.id
http://192.168.111.10:8100 0
4 native.pbjtservers.my.id
http://192.168.111.10:8400 0
```

Gambar 3. 3 Public Hostname

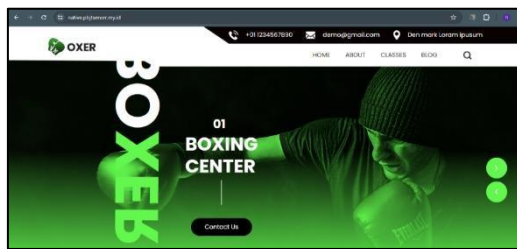
Pengujian

Tahapan pengujian bertujuan untuk menentukan apakah sistem yang dibuat telah sesuai dengan prosedur yang telah ditentukan dan dapat berjalan dengan baik. Pengujian dilakukan menggunakan *Apache JMeter* untuk mengukur beban, waktu *respons*, keandalan, pemulihan, dan skalabilitas *web server*.

HASIL DAN PEMBAHASAN

Hasil

A. Hasil Tampilan Website Native



Gambar 3. 4 Tampilan Website Native

B. Hasil Tampilan Website Wordpress



Gambar 3. 5 Tampilan Website Wordpress

C. Hasil Pengujian

Pengujian melibatkan parameter seperti waktu *respons*, keandalan, pemulihan, dan skalabilitas. Sepuluh pengguna mengakses web server secara bersamaan untuk memastikan hasil representatif. Berikut adalah contoh hasil pengujiannya:

Tabel 4. 1 Hasil Pengujian

	Native	Wordpress
Uji Beban	623ms	1798ms
Waktu Respons	2.6ms	1.6ms
Keandalan	✓	✓
Pemulihan	✓	✓
Skalabilitas	✓	✓

Pembahasan

hasil implementasi *web server* berbasis *CasaOS* menggunakan *Docker* dan *Cloudflare Tunnels* pada sistem operasi *Linux Ubuntu*. Penelitian ini bertujuan untuk menciptakan *web server* yang mudah diakses dari internet secara gratis dan hemat biaya, serta bermanfaat untuk media publikasi. Hasil implementasi menunjukkan bahwa *website native* dan *website* berbasis *WordPress* berhasil dikonfigurasi dan diakses dengan baik melalui *web server* yang diimplementasikan. Kedua jenis *website* menunjukkan performa yang stabil dan responsif, dengan waktu respon masing-masing 623ms dan 1798ms pada pengujian beban. Waktu *respons website* native adalah 2.6 detik, sedangkan *WordPress* 1.6 detik, menunjukkan efisiensi dan kemudahan akses. Kedua *website* juga menunjukkan keandalan yang baik dan waktu pemulihan yang cepat setelah gangguan, serta mampu menangani peningkatan jumlah pengguna tanpa penurunan performa yang signifikan.

Kelebihan dan Kekurangan

Penelitian ini memiliki beberapa kelebihan, termasuk efisiensi biaya, kontrol penuh atas data dan sistem, kemudahan penggunaan, isolasi aplikasi yang baik melalui *Docker*, dan keamanan tambahan dari *Cloudflare Tunnels*. Namun, penelitian ini juga memiliki kekurangan seperti keterbatasan hardware yang mempengaruhi performa *web server*, kompleksitas implementasi yang membutuhkan pemahaman teknis yang mendalam, keterbatasan lingkup penelitian yang hanya fokus pada dua platform website dan satu sistem operasi, serta ketergantungan pada konektivitas internet.

KESIMPULAN DAN SARAN

Kesimpulan

Implementasi *web server* berbasis *CasaOS* menggunakan *Docker* dan *Cloudflare Tunnels* pada sistem operasi *Linux Ubuntu* telah berhasil menciptakan *platform* yang efisien dan

ekonomis. *CasaOS* menyediakan antarmuka grafis yang mudah digunakan, *Docker* memastikan isolasi aplikasi, dan *Cloudflare Tunnels* menawarkan koneksi terenkripsi yang aman antara *server* lokal dan internet. Hasil implementasi menunjukkan bahwa *web server* ini mudah dikelola, aman, dan dapat diakses dari internet tanpa biaya tambahan.

Saran

1. Tingkatkan CPU, RAM, dan penyimpanan untuk beban kerja berat.
2. Gunakan *Ansible* atau *Terraform*, serta pemantauan *real-time* seperti *Prometheus* dan *Grafana*.
3. Implementasikan *Zero Trust Architecture* dan *firewalls* canggih.
4. Terapkan *load balancing*, *caching*, dan gunakan *Kubernetes* untuk manajemen kontainer.
5. Gunakan *microservices*, *serverless architecture*, dan *API Gateway* untuk fleksibilitas dan keamanan.

DAFTAR PUSTAKA

Ariadi, F. et al. (2020). Penerapan Docker Container Sebagai Teknologi Ramah Skalabilitas Dibanding Teknik Virtualisasi Untuk Membangun Website Di Ubuntu 18.04.4 Lts. *Jurnal JARKOM*, 8(2), 47–57.

Boras, M. et al. (2020). Performance Evaluation of Linux Operating Systems. *Proceedings of 2020 International Conference on Smart Systems and Technologies, SST 2020*, (October 2020), 115–120. <https://doi.org/10.1109/SST49455.2020.9264055>

Daheri, M. (2023). Implementation of Management Information Systems at State Junior High School 2 Rejang Lebong Bengkulu. *EDUKASI : Jurnal Pendidikan Islam (e-Journal)*, 11(1), 88–97. <https://doi.org/10.54956/edukasi.v11i1.373>

Györödi, C. A. et al. (2021). Performance

impact of optimization methods on MySQL document-based and relational databases. *Applied Sciences (Switzerland)*, 11(15). <https://doi.org/10.3390/app11156794>

Ibrahim, A. M. (2023). Program studi teknik multimedia dan jaringan jurusan teknik informatika dan komputer politeknik negeri jakarta 2023, (1907423006).

Ihsan et al. (2023). Perancangan Infrastruktur Dan Implementasi Web Server Untuk Website Sekolah Sebagai Media Informasi Dan Komunikasi Di Smp Pjhi Balikpapan. *Jurnal Teknologi Terpadu*, 11(1), 66–72. <https://doi.org/10.32487/jtt.v11i1.1598>

Kesuma Astuti, F. et al. (2022). Membangun Website MTS Negeri 01 OKU Timur Menggunakan Php dan Mysql. *Jik*, 13(1), 7–14.

Kostrzewa-Nowak, D. et al. (2023). Practical Applications of Target Costing in a Multidisciplinary R&D Project. *Sustainability (Switzerland)*, 15(1), 1–13. <https://doi.org/10.3390/su15010124>

Maesaroh, S. et al. (2022). Query Optimization In MySQL Database Using Index. *International Journal of Cyber and IT Service Management*, 2(2), 104–110. <https://doi.org/10.34306/ijcitsm.v2i2.84>

Mangayarkarasi, M. et al. (2021). Highly Scalable And Load Balanced Web Server On Aws Cloud. *IOP Conference Series: Materials Science and Engineering*, 1055(1), 012113. <https://doi.org/10.1088/1757-899x/1055/1/012113>

Miell, I. et al. (2019). *Docker in Practice*. Manning Publications.

Rose, S. et al. (2022). Zero Trust Architecture. *Controlling Privacy and the Use of Data Assets*, 127–134.

<https://doi.org/10.1201/9781003189664-11>

- Saraswati, I. et al. (2019). Increasing Web Server Performance Using The Web Balancing Method. *IOP Conference Series: Materials Science and Engineering*, 673(1). <https://doi.org/10.1088/1757-899X/673/1/012065>
- Sinha, V. A. et al. (2023). Domain Analysis and Feature Identification of Virtualization in Debian Linux Using Type II Hypervisor-Virtual Box. *International Journal of Computer Science Trends and Technology*, 11(4), 22–30. Retrieved from www.ijcstjournal.org
- Susilo, I. et al. (2019). Pembangunan Web Server Menggunakan Debian Server Untuk Media Pembelajaran Di Sekolah Menengah Kejuruan (Smk) Negeri 1 Sragen. *Indonesian Jurnal on Networking and Security (IJNS)-Ijns.Org IJNS*, 2(1), 2302–5700. Retrieved from <http://kuis.smkn1srg.sch.id>
- Tech, I. (2024). *Your Personal Cloud OS*. Retrieved from <https://www.casaos.io/>