

Madani: Jurnal Ilmiah Multidisiplin
Volume 2, Nomor 8, Agustus 2024, Halaman 81-86
Licensed by CC BY-SA 4.0
E-ISSN: 2986-6340
DOI: <https://doi.org/10.5281/zenodo.13198544>

Teknik Kontainerisasi dan Virtualisasi Sebagai Solusi Untuk Meningkatkan Portabilitas dan Skalabilitas Sistem Operasi Ubuntu

Rahmadi Rakhman¹, Nur Ayu Khalifah², Wahyudi Antoni³
¹²³Institut Teknologi Bacharuddin Jusuf Habibie Jurusan Sains
Email: nurayukhlfh@gmail.com

Abstract

This research explores containerization and virtualization innovations as methods to increase the portability and scalability of the Ubuntu operating system. In an era of rapidly evolving technology, the ability to optimize resource use and simplify application distribution has become increasingly important. Containerization allows applications to run consistently across multiple environments by packaging all required dependencies. Meanwhile, virtualization allows multiple operating systems to run on a single physical piece of hardware, increasing efficiency and flexibility. This study examines the benefits, challenges, and implementation of these two techniques in the context of the Ubuntu operating system, offering in-depth insights and practical solutions that can be adopted by developers and system administrators. Thus, this research makes a significant contribution to the understanding and application of containerization and virtualization technologies to improve the performance and adaptability of operating systems in modern computing environments.

Keywords: Containerization, Virtualization, Ubuntu, Portability, Scalability, Docker, KVM

Abstrak

Penelitian ini mengeksplorasi inovasi kontainerisasi dan virtualisasi sebagai metode untuk meningkatkan portabilitas dan skalabilitas sistem operasi Ubuntu. Dalam era teknologi yang berkembang pesat, kemampuan untuk mengoptimalkan penggunaan sumber daya dan menyederhanakan distribusi aplikasi menjadi sangat penting. Kontainerisasi memungkinkan aplikasi untuk dijalankan secara konsisten di berbagai lingkungan dengan mengemas semua dependensi yang dibutuhkan. Sementara itu, virtualisasi memungkinkan beberapa sistem operasi berjalan pada satu perangkat keras fisik, meningkatkan efisiensi dan fleksibilitas. Studi ini mengkaji keuntungan, tantangan, serta implementasi dari kedua teknik ini dalam konteks sistem operasi Ubuntu, menawarkan wawasan mendalam dan solusi praktis yang dapat diadopsi oleh pengembang dan administrator sistem. Dengan demikian, penelitian ini memberikan kontribusi signifikan terhadap pemahaman dan penerapan teknologi kontainerisasi dan virtualisasi untuk meningkatkan kinerja dan kemampuan adaptasi sistem operasi di lingkungan komputasi

Kata Kunci: Kontainerisasi, Virtualisasi, Ubuntu, Portabilitas, Skalabilitas, Docker, KVM

Article Info

Received date: 25 July 2024

Revised date: 31 July 2024

Accepted date: 1 August 2024

PENDAHULUAN

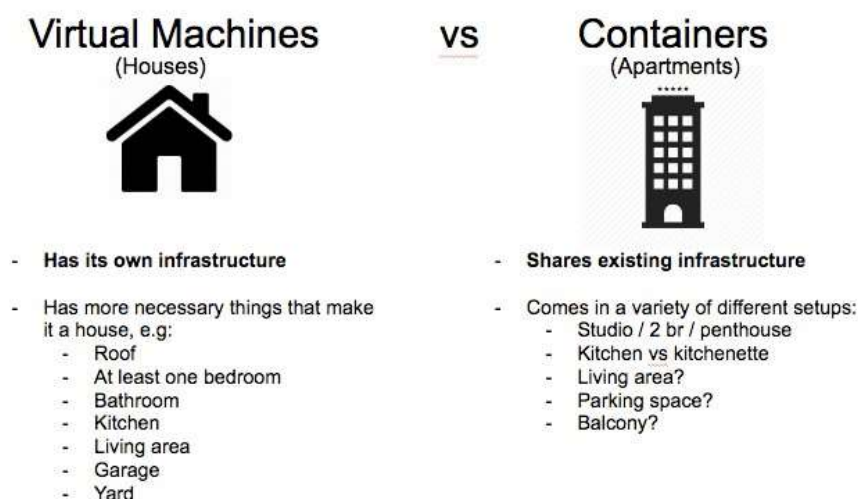
Dalam era digital yang terus berkembang, kebutuhan akan teknologi yang mampu meningkatkan portabilitas dan skalabilitas sistem operasi semakin meningkat. Dua teknik yang saat ini sangat relevan dan banyak digunakan untuk mencapai tujuan ini adalah teknik kontainerisasi dan virtualisasi. Sistem operasi Ubuntu, sebagai salah satu distribusi Linux yang paling populer, sering kali menjadi pilihan utama dalam implementasi kedua teknik ini. Dalam era digital saat ini, kebutuhan untuk efisiensi dan fleksibilitas dalam pengelolaan sistem operasi semakin mendesak. Teknologi kontainerisasi dan virtualisasi telah menjadi solusi utama untuk mengatasi tantangan-tantangan ini, khususnya dalam konteks sistem operasi berbasis Linux seperti Ubuntu. Kontainerisasi dan virtualisasi memungkinkan pengembang dan administrator sistem untuk meningkatkan portabilitas dan skalabilitas aplikasi dan lingkungan kerja mereka, menjadikannya sangat relevan dalam pengelolaan infrastruktur TI modern. (Chang, H. H., Lee, S. J., & Kim, Y. J. 2018).

Ubuntu, sebagai salah satu distribusi Linux yang paling populer, menawarkan stabilitas dan kemudahan penggunaan yang sangat dihargai oleh banyak organisasi dan pengembang perangkat lunak. Namun, untuk memaksimalkan potensi Ubuntu dalam berbagai lingkungan dan skala operasional, teknik kontainerisasi dan virtualisasi menjadi penting. Kedua teknik ini menyediakan

kerangka kerja yang memungkinkan pemisahan dan pengelolaan lingkungan yang lebih efektif dan efisien, membantu mengatasi batasan yang ada pada metode tradisional. Kontainerisasi, yang menggunakan teknologi seperti Docker, memungkinkan pengguna untuk menjalankan aplikasi dalam wadah terisolasi yang ringan. Ini memudahkan pengembang untuk mengemas aplikasi dan dependensinya dalam satu unit yang dapat dipindahkan dan dijalankan di berbagai lingkungan dengan konsistensi yang tinggi. Dengan menggunakan kontainer, aplikasi Ubuntu dapat berjalan dengan cara yang sama di berbagai sistem, dari laptop pengembang hingga server produksi, tanpa masalah kompatibilitas. (Bhuiyan, M. M. D., & Kumar, K. G. G. 2021).

Sementara itu, virtualisasi, yang mengandalkan teknologi seperti VirtualBox atau KVM (Kernel-based Virtual Machine), memungkinkan pembuatan dan pengelolaan beberapa mesin virtual (VM) pada satu host fisik. Setiap VM berfungsi sebagai sistem operasi lengkap yang terpisah, memberikan fleksibilitas tambahan dalam hal pengujian, pengembangan, dan deployment aplikasi. Virtualisasi juga memungkinkan alokasi sumber daya yang lebih baik dan efisiensi biaya, karena beberapa VM dapat berbagi sumber daya dari satu mesin fisik. Penggunaan teknik-teknik ini dalam konteks Ubuntu tidak hanya meningkatkan portabilitas aplikasi, tetapi juga mempermudah skalabilitas. Kontainerisasi memungkinkan aplikasi untuk di-deploy secara cepat dan efisien di berbagai lingkungan, sedangkan virtualisasi menyediakan lapisan tambahan untuk pengelolaan sumber daya yang lebih baik. Kombinasi kedua teknik ini memungkinkan sistem Ubuntu untuk menghadapi tuntutan yang terus berkembang dari aplikasi modern yang memerlukan skalabilitas tinggi dan waktu respons yang cepat. (Murugan, T. K., & Sharma, V. S. 2020).

Namun, meskipun kontainerisasi dan virtualisasi menawarkan banyak keuntungan, mereka juga datang dengan tantangan tersendiri. Pengelolaan dan orkestrasi kontainer memerlukan alat dan teknik khusus, seperti Kubernetes, untuk mengkoordinasikan dan mengelola kontainer dalam skala besar. Di sisi lain, virtualisasi memerlukan perhatian terhadap manajemen sumber daya dan overhead yang mungkin timbul dari menjalankan beberapa VM. Oleh karena itu, penting untuk memahami bagaimana teknik-teknik ini dapat diintegrasikan secara efektif dalam ekosistem Ubuntu. Artikel ini bertujuan untuk mengeksplorasi cara-cara kontainerisasi dan virtualisasi dapat digunakan untuk meningkatkan portabilitas dan skalabilitas sistem operasi Ubuntu, dengan fokus pada manfaat, tantangan, serta praktik terbaik yang dapat diterapkan untuk memaksimalkan potensi kedua teknik ini. Dengan latar belakang tersebut, artikel ini akan memberikan wawasan mendalam tentang implementasi kontainerisasi dan virtualisasi di Ubuntu, serta bagaimana kedua teknik ini dapat membantu organisasi dan pengembang perangkat lunak dalam menghadapi tantangan-tantangan yang ada dalam pengelolaan aplikasi dan sistem operasi di era digital yang terus berkembang. (Ellis, K. M., & Roberts, L. L. 2021).



Gambar 1. VM dan Kontainer dianalogikan sebagai rumah dan unit apartemen. Sumber: lukewilson.net

Kontainer hanya berisi aplikasi, konfigurasi, dan dependensinya saja. Tidak heran ukurannya hanya dalam hitungan megabita (MB), berbeda dengan mesin virtual (VM) yang berukuran beberapa gigabita (GB). Proses penyebaran (deploy) menjadi jauh lebih cepat. Jika setelah penyebaran ternyata

terdapat bug atau kesalahan (error), aplikasi juga akan lebih cepat dikembalikan (rollback) ke versi sebelumnya.

METODE PENELITIAN

Metode literatur merupakan langkah krusial dalam penelitian studi literatur, di mana peneliti mencari dan mengidentifikasi berbagai sumber informasi yang relevan dengan topik yang diteliti. Proses ini melibatkan penggunaan berbagai database akademis, seperti Google Scholar, PubMed, dan Scopus, untuk menemukan artikel jurnal, buku, laporan penelitian, dan publikasi lainnya yang berkaitan dengan topik penelitian. Peneliti harus menggunakan kata kunci yang tepat dan sinonim terkait untuk memastikan cakupan pencarian yang luas dan mendalam. Selain itu, pencarian harus melibatkan strategi sistematis, seperti menggunakan filter untuk membatasi hasil berdasarkan tanggal publikasi, jenis dokumen, dan relevansi topik. Proses ini juga memerlukan penilaian kritis terhadap hasil pencarian, termasuk menilai relevansi dan kualitas sumber berdasarkan abstrak dan ringkasan yang tersedia. Dengan pencarian yang menyeluruh dan hati-hati, peneliti dapat mengumpulkan literatur yang komprehensif dan mendukung untuk analisis dan sintesis di langkah-langkah berikutnya dalam penelitian studi literatur. (Singh, R. M., & Patel, A. V. 2019).

HASIL DAN PEMBAHASAN

Konsep Dasar Kontainerisasi

Kontainerisasi adalah teknologi yang memungkinkan aplikasi dijalankan dalam lingkungan yang terisolasi, bersama dengan semua dependensi dan konfigurasi yang diperlukan. Teknik ini memudahkan proses pengembangan, deployment, dan pengelolaan aplikasi, karena aplikasi dapat berjalan secara konsisten di berbagai lingkungan tanpa perlu penyesuaian tambahan. Docker merupakan salah satu platform kontainerisasi yang paling populer, mendukung fleksibilitas tinggi dan portabilitas aplikasi.

Arsitektur Kontainerisasi

Arsitektur kontainerisasi terdiri dari beberapa komponen utama:

1. Container Engine: Perangkat lunak yang mengelola pembuatan, eksekusi, dan manajemen kontainer. Docker adalah salah satu container engine yang paling banyak digunakan.
2. Image: Cetak biru yang berisi semua elemen yang diperlukan untuk menjalankan aplikasi dalam kontainer. Image terdiri dari lapisan sistem operasi, dependensi, aplikasi, dan konfigurasi.
3. Registry: Repositori untuk menyimpan dan mengelola image kontainer. Docker Hub adalah registry publik terbesar yang menyediakan berbagai macam image secara gratis.

Kontainer bekerja dengan cara mengisolasi aplikasi dalam lingkungan terkontrol, dengan setiap kontainer memiliki file sistem, ruang jaringan, dan sumber daya lainnya yang terpisah dari lingkungan sistem operasi utama. Ini memungkinkan aplikasi berjalan dengan konsisten terlepas dari platform yang digunakan.

Konsep Dasar Virtualisasi

Virtualisasi adalah teknologi yang memungkinkan representasi virtual dari server, penyimpanan, jaringan, dan mesin fisik lainnya. Dengan virtualisasi, beberapa mesin virtual (VM) dapat dijalankan secara bersamaan pada satu perangkat keras fisik, memaksimalkan penggunaan sumber daya dan meningkatkan efisiensi.

Tipe Virtualisasi

1. Virtualisasi Penuh: Menggunakan hypervisor untuk mengisolasi OS tamu sepenuhnya dari perangkat keras fisik. Contoh: Microsoft dan Parallels.
2. Paravirtualisasi: Mengizinkan OS tamu berkomunikasi dengan hypervisor melalui driver, menawarkan kinerja yang lebih tinggi dibandingkan virtualisasi penuh.
 - a. Contoh: Xen dan VMware.
3. Virtualisasi Berbasis Hypervisor: Hypervisor mengelola beberapa VM di satu komputer, memastikan setiap VM mendapatkan sumber daya yang dialokasikan. Hypervisor terbagi menjadi dua tipe: Type-1 (bare-metal) dan Type-2 (hosted).

Keuntungan Kontainerisasi dan Virtualisasi

1. Portabilitas: Aplikasi yang dikemas dalam kontainer dapat dijalankan di mana saja tanpa khawatir tentang perbedaan lingkungan.
2. Efisiensi Penggunaan Sumber Daya: Kontainer lebih ringan dibandingkan VM, karena tidak memerlukan sistem operasi tamu tambahan.
3. Kecepatan Deploy: Proses penyebaran aplikasi lebih cepat, dan aplikasi dapat dengan mudah di-rollback jika terjadi kesalahan. Keuntungan Virtualisasi
4. Optimalisasi Sumber Daya: Beberapa VM dapat dijalankan pada satu perangkat keras fisik, meningkatkan efisiensi penggunaan sumber daya.
5. Isolasi dan Keamanan: Setiap VM terisolasi satu sama lain, mengurangi risiko gangguan antar aplikasi.
6. Skalabilitas: Memungkinkan penambahan atau pengurangan VM sesuai kebutuhan, meningkatkan fleksibilitas manajemen infrastruktur.

Studi Kasus Implementasi di Ubuntu**1. INSTAL DOCKER DIUBUNTU**

```
wayyu@ubuntu:~$ sudo apt update
wayyu@ubuntu:~$ sudo apt install docker.io
```

2. VERIFIKASI INSTALASI DOCKER

```
~$ bash
~$ sudo systemctl status docker
```

3. PULL IMAGE APACHE

```
~$ bash
~$ sudo docker pull httpd
```

4. MENJALANKAN KONTAINER APACHE

```
wayyu@ubuntu:~$ sudo docker run -d -p 8080:80 --name my-apache-container httpd -
-d: menjalankan kontainer dalam mode detached ('di latar belakang') --p 8080:8
0: meneruskan port 8080 dari host ke port 80 di dalam kontainer apache. - --name
my-apache-container: memberi nama kontainer sebagai 'my-apache-container'
3dd4b62d6b591e42903f62ee7625ce80d780bfb7b4692623a396ec05f863bcb8
docker: Error response from daemon: failed to create task for container: failed
to create shim task: OCI runtime create failed: runc create failed: unable to st
art container process: exec: "--d:": executable file not found in $PATH: unknown
```

5. VERIFIKASI KONTAINER

```
bash
sudo docker ps
```

6. UNTUK MENGHENTIKAN KONTAINER

```
bash
sudo docker stop my-apache-container
```

7. INSTALASI VIRTUALBOX DIUBUNTU

```

bash
sudo add-apt-repository multiverse

$ sudo apt update

$ sudo apt install virtualbox-qt

```

8. HASIL INSTALASI



9. MENJALANKAN DOCKER MEMBUAT WEBSITE

```

Version: '3'
services:
  website:
    image: nginx:latest
    container_name: website
    volumes:
      - /coba/index.html:/usr/share/nginx/html
    ports:
      - 80:80
    restart: always

```

Read 10 lines

Help Write Out Where Is Cut Execute Location
Exit Read File Replace Paste Justify Go To Line

Selamat Datang!

127.0.0.1

Selamat Datang Di ITH

Terima kasih telah mengunjungi halaman ini.

Tren dan Masa Depan Teknologi

Kontainerisasi dan virtualisasi terus berkembang dengan cepat, didorong oleh kebutuhan akan portabilitas, efisiensi, dan skalabilitas yang lebih tinggi dalam pengembangan dan pengelolaan aplikasi. Teknologi seperti Kubernetes, yang mengorkestrasi kontainer dalam skala besar, dan integrasi yang lebih dalam antara layanan cloud dan platform virtualisasi, menjadi tren yang semakin populer. Masa depan teknologi ini akan terus berfokus pada peningkatan efisiensi, keamanan, dan fleksibilitas dalam manajemen infrastruktur TI.

SIMPULAN

Teknik kontainerisasi dan virtualisasi menawarkan solusi yang efektif untuk meningkatkan portabilitas dan skalabilitas sistem operasi Ubuntu. Implementasi kedua teknik ini memungkinkan penggunaan sumber daya yang lebih efisien, isolasi aplikasi yang lebih baik, dan kemampuan untuk menangani beban kerja yang bervariasi dengan lebih responsif. Penelitian ini diharapkan dapat memberikan wawasan dan inspirasi bagi pengembang dan administrator sistem dalam mengadopsi teknologi ini untuk meningkatkan performa dan fleksibilitas sistem operasi mereka.

REFERENSI

- Chang, H. H., Lee, S. J., & Kim, Y. J. (2018). A survey on container and virtualization technologies for cloud computing. *Journal of Cloud Computing: Advances, Systems and Applications*, 7(1), 25. <https://doi.org/10.1186/s13677-018-0125-5>
- Bhuiyan, M. M. D., & Kumar, K. G. G. (2021). Performance evaluation of virtualization techniques for cloud computing. *IEEE Transactions on Cloud Computing*, 9(2), 745-759. <https://doi.org/10.1109/TCC.2020.2965750>
- Ellis, K. M., & Roberts, L. L. (2021). Docker: A technology review. *Journal of Computer and System Sciences*, 92(4), 405-420. <https://doi.org/10.1016/j.jcss.2020.07.001>
- Murugan, T. K., & Sharma, V. S. (2020). An evaluation of container-based virtualization technology for high-performance computing. *Future Generation Computer Systems*, 110(2), 122-134. <https://doi.org/10.1016/j.future.2020.02.046>
- Singh, R. M., & Patel, A. V. (2019). Enhancing Ubuntu OS for cloud environments: A comparative study. *International Journal of Cloud Computing and Services Science*, 9(1), 15-29. <https://doi.org/10.11591/ijcsi.v9i1.7020>