

Analisis Performa Virtualisasi Sistem Operasi Windows 8 Menggunakan Oracle VM VirtualBox dengan Variasi Konfigurasi RAM

Sakhrani Rasyid¹, Sulistia², Aidil Fajri Adinata³, Riko Taruna⁴, Sahat Parulian Sitorus⁵

¹ program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Labuhanbatu, Rantauprapat, Indonesia

² program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Labuhanbatu, Rantauprapat, Indonesia

³ program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Labuhanbatu, Rantauprapat, Indonesia

⁴ program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Labuhanbatu, Rantauprapat, Indonesia

⁵ program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Labuhanbatu, Rantauprapat, Indonesia

Info Artikel

Riwayat Artikel:

Diterima 23, Juni, 2026

Perbaikan 30, Juni, 2026

Disetujui 17, Juli, 2026

Keywords:

Windows 8, Virtualisasi,
Oracle VM VirtualBox,
Alokasi Sumber Daya,
Analisis Performa Komparatif.

ABSTRAK

Sistem operasi mengelola sumber daya perangkat keras dan menyediakan antarmuka pengguna. Windows 8 masih sangat relevan untuk dikaji karena mewakili transisi arsitektur yang krusial dari antarmuka desktop klasik (Windows 7) ke antarmuka modern berbasis tile (Windows 10/11). Karakteristik transisional grafis (Metro UI) ini menjadikannya objek uji beban yang ideal bagi hypervisor tipe 2 dalam menangani sistem operasi guest.

Berbeda dengan penelitian terdahulu yang umumnya mengandalkan konfigurasi tunggal, penelitian deskriptif-kuantitatif ini mengevaluasi performa instalasi dan konsumsi sumber daya Windows 8 pada Oracle VM VirtualBox melalui skema multi-konfigurasi. Pengujian membandingkan alokasi RAM guest sebesar 2 GB dan 4 GB, di mana setiap skenario direplikasi tiga kali guna mendapatkan nilai rata-rata yang valid. Performa host dipantau secara simultan selama instalasi menggunakan Windows Task Manager dan Resource Monitor.

Hasil pengujian menunjukkan perbedaan performa yang signifikan. Konfigurasi RAM 2 GB mencatatkan rata-rata durasi instalasi 20,5 menit dengan puncak utilitas CPU host 93,4%, serta ukuran aktual Virtual Disk Image (VDI) pasca-instalasi 12,5 GB. Sementara itu, konfigurasi RAM 4 GB lebih efisien dengan durasi 17,2 menit, puncak CPU 89,1%, dan ukuran VDI 12,8 GB. Data ini membuktikan bahwa penambahan RAM guest mampu mempercepat proses paging dan menekan beban kerja subsistem memori virtual, meskipun beban CPU tetap dominan pada fase dekompresi berkas biner.

Penelitian ini memberikan kontribusi berupa data komparatif kuantitatif yang objektif. Temuan tersebut dapat diimplementasikan sebagai rujukan praktis bagi para pengajar dan mahasiswa untuk menentukan alokasi sumber daya minimum yang paling optimal saat menjalankan praktikum virtualisasi sistem operasi legacy pada infrastruktur laboratorium yang terbatas.

ABSTRACT

Operating systems play a crucial role in managing hardware resources and providing user interfaces. Windows 8 remains highly relevant for study because it represents a critical architectural transition from the classic desktop interface (Windows 7) to the modern tile-based interface (Windows 10/11). These transitional graphical characteristics (Metro UI) make it an ideal load-testing object for a type 2 hypervisor in managing guest operating systems.

Unlike previous virtualization studies that generally relied on a single configuration, this descriptive-quantitative research evaluates the installation performance and resource consumption of Windows 8 on Oracle VM VirtualBox through a multi-configuration scheme. The test compares guest RAM allocations of 2 GB and 4 GB, with each scenario replicated three times to obtain a valid mean value. Host performance was monitored simultaneously throughout the installation process using Windows Task Manager and Resource Monitor.

The test results demonstrate significant differences in performance metrics. The 2 GB RAM configuration recorded an average installation duration of 20.5 minutes, a peak host CPU utilization of 93.4%, and an actual post-installation Virtual Disk Image (VDI) footprint of 12.5 GB. Meanwhile, the 4 GB RAM configuration proved more efficient with an average duration of 17.2 minutes, a peak CPU utilization of 89.1%, and a VDI size of 12.8 GB. This data indicates that increasing the guest RAM effectively accelerates the paging process and reduces the workload of the virtual memory subsystem, although the CPU load remains dominant during the binary file decompression phase.

This study contributes objective, quantitative comparative data. These findings can be implemented as a practical reference for educators and students to determine the optimal minimum resource allocation when conducting legacy operating system virtualization practical sessions in laboratory infrastructures with limited hardware resources.

Ini adalah artikel akses terbuka di bawah lisensi CC BY-SA.



1. PENDAHULUAN

Perkembangan teknologi informasi dan komputer pada era digital menunjukkan pertumbuhan yang sangat pesat. Sistem operasi sebagai perangkat lunak fundamental memainkan peranan yang sangat penting dalam setiap komputer modern [1]. Sistem operasi berfungsi sebagai perantara (*interface*) antara pengguna dan perangkat keras komputer, serta mengelola seluruh sumber daya *hardware* termasuk *processor*, memori, *storage*, dan perangkat *input-output* [4]. Tanpa kehadiran sistem operasi, pengguna tidak dapat berkomunikasi dengan perangkat keras komputer secara efektif [6]. Keluarga sistem operasi Windows yang dikembangkan oleh Microsoft merupakan salah satu platform yang paling banyak diimplementasikan di seluruh dunia [6]. Windows 8 menghadirkan perubahan paradigma dalam antarmuka pengguna dengan memperkenalkan *Metro UI* dan dukungan penuh untuk perangkat layar sentuh [2], [9].

Meskipun Windows 8 menawarkan inovasi berupa *booting* yang lebih cepat melalui optimasi arsitektur *Unified Extensible Firmware Interface* (UEFI) dan sistem keamanan terintegrasi seperti *Secure Boot* [5], sistem ini juga memberikan tantangan berupa *learning curve* yang curam bagi pengguna tradisional akibat penghapusan *Start Menu* konvensional [10]. Dalam konteks akademis dan pengujian infrastruktur IT, melakukan instalasi atau eksperimen langsung pada perangkat keras fisik memiliki risiko tinggi, seperti kehilangan data atau kerusakan sistem operasi utama *host* [15]. Oleh karena itu, diperlukan sebuah mekanisme isolasi lingkungan kerja yang aman guna mengevaluasi karakteristik arsitektur sistem operasi *guest* secara komprehensif.

Teknologi virtualisasi hadir sebagai solusi krusial yang memungkinkan eksekusi beberapa mesin virtual (*virtual machine*) secara independen di atas satu infrastruktur fisik yang sama [7]. Oracle VM VirtualBox merupakan salah satu aplikasi *hypervisor* tipe 2 berbasis *open-source* yang banyak dimanfaatkan karena fleksibilitasnya dalam mendukung berbagai sistem operasi *guest* serta dukungannya terhadap arsitektur *full virtualization* [3]. Penelitian terdahulu menunjukkan bahwa

pemanfaatan VirtualBox sangat efektif sebagai media pembelajaran praktikum karena menyediakan lingkungan simulasi yang representatif tanpa membebani biaya infrastruktur tambahan [24], [25].

Pemilihan Windows 8 sebagai objek penelitian bukan tanpa alasan. Pertama, dari sudut pandang akademis, Windows 8 merepresentasikan titik transisi arsitektural yang unik antara model antarmuka desktop klasik pada Windows 7 dan model antarmuka *tile-based* (Metro UI) yang kemudian disempurnakan pada Windows 10/11, sehingga mempelajari perilaku instalasi dan konsumsi sumber dayanya memberikan gambaran empiris mengenai beban komputasi yang ditimbulkan oleh perubahan paradigma antarmuka tersebut di dalam lingkungan tervirtualisasi. Kedua, dari sudut pandang praktis, Windows 8 dan Windows 8.1 masih ditemukan beroperasi pada sejumlah sistem *legacy*, perangkat *embedded*, serta laboratorium pendidikan vokasi yang belum memperbarui citra sistemnya, sehingga kajian mengenai kebutuhan sumber daya minimum yang realistis untuk menjalankannya secara virtual tetap memiliki nilai kegunaan bagi pengelola laboratorium dengan anggaran terbatas.

Sejumlah penelitian terdahulu telah mengkaji performa *hypervisor* tipe 2, namun umumnya berfokus pada perbandingan antar-produk *hypervisor* dan bukan pada variasi konfigurasi sumber daya untuk satu sistem operasi *guest* tertentu. Vojnak dkk. [26] membandingkan performa Oracle VM VirtualBox dengan VMware Workstation pada skenario satu, dua, dan tiga mesin virtual yang berjalan bersamaan, namun tidak menyertakan variasi alokasi RAM *guest* tunggal maupun sistem operasi *guest* berbasis Windows 8. Penelitian lain menguji performa berbagai sistem *file guest* pada VirtualBox menggunakan *benchmark Filebench* [27], serta kajian mengenai performa mesin virtual Windows pada *host* berbasis Linux [28] dan evaluasi kritis performa *hypervisor* menggunakan tolok ukur *disk I/O*, memori, *CPU throughput*, dan konsumsi energi [29]. Ketiadaan penelitian yang secara spesifik memvariasikan alokasi RAM *guest* pada satu sistem operasi tetap (Windows 8) dengan pengulangan eksperimen untuk memperoleh nilai rata-rata menjadikan celah (“gap”) penelitian yang coba diisi oleh studi ini.

Berdasarkan permasalahan dan celah penelitian tersebut, penelitian ini bertujuan untuk mengevaluasi secara eksperimental proses instalasi Windows 8 pada VirtualBox pada dua konfigurasi alokasi RAM *guest* (2 GB dan 4 GB), menganalisis metrik performa meliputi durasi instalasi, efisiensi ukuran berkas *virtual disk*, dan beban utilitas sumber daya *hardware* pada *host*, serta memberikan rekomendasi praktis mengenai alokasi sumber daya minimum yang optimal untuk kebutuhan praktikum sistem operasi *legacy* di lingkungan laboratorium akademik.

2. METODE

Penelitian ini menerapkan metode penelitian deskriptif eksperimental berbasis pengujian laboratorium langsung dengan desain *multi-configuration testing* (skenario eksperimen multi-konfigurasi). Berbeda dari rancangan awal yang hanya menguji satu konfigurasi *guest* tunggal, penelitian ini membangun dua profil mesin virtual dengan alokasi RAM *guest* yang berbeda, yaitu 2 GB dan 4 GB, sementara parameter lain (jumlah *virtual CPU*, kapasitas *virtual disk*, mode jaringan) dipertahankan tetap (dikontrol) agar pengaruh RAM terhadap performa dapat diisolasi secara lebih valid. Setiap konfigurasi dijalankan sebanyak **tiga kali pengulangan (replikasi, n = 3)** pada kondisi *host* yang identik (aplikasi latar belakang ditutup, *host* dalam kondisi *idle* sebelum pengujian dimulai), dan hasil yang dilaporkan merupakan nilai rata-rata (*mean*) dari ketiga pengulangan tersebut guna meningkatkan validitas dan keterulangan (*reproducibility*) data.

Pemantauan konsumsi sumber daya *host* dilakukan secara simultan menggunakan dua alat ukur bawaan sistem operasi *host*, yaitu *Windows Task Manager (tab Performance)* untuk memantau utilitas CPU dan RAM secara *real-time*, serta *Resource Monitor (Performance Monitor)* untuk mencatat rincian penggunaan memori dan aktivitas *disk I/O* per proses VBoxHeadless/VirtualBoxVM.exe. Pembacaan nilai persentase CPU dan RAM dilakukan pada interval setiap 30 detik selama durasi instalasi berlangsung, kemudian nilai puncak (*peak*) dan nilai rata-rata dari seluruh pembacaan dicatat untuk setiap sesi. Durasi instalasi diukur menggunakan *stopwatch* digital sejak proses *boot* pertama dari media ISO hingga sistem *guest* mencapai tampilan *desktop* pasca-konfigurasi akun lokal. Ukuran berkas *virtual disk image* (VDI) diperiksa langsung melalui *File Explorer* pada direktori penyimpanan mesin virtual *host* setelah proses instalasi selesai.

Alat dan bahan utama yang digunakan dalam penelitian meliputi perangkat keras laptop, *hypervisor* Oracle VM VirtualBox, serta berkas *image* resmi Windows 8 Pro dalam format ISO.

Spesifikasi teknis lingkungan *host* dan rancangan alokasi parameter pada mesin virtual *guest* untuk kedua skenario disajikan secara komprehensif pada Tabel 1.

Tabel 1. Spesifikasi Perangkat Keras Host dan Alokasi Parameter Mesin Virtual (Guest)

Parameter	Host	Guest – Konfigurasi A (2 GB)	Guest – Konfigurasi B (4 GB)
Processor	[Intel Core i3-10110U, 2 Core / 4 Thread]	Virtual CPU: 1 Core	Virtual CPU: 1 Core
RAM / Memori Utama	8 GB DDR4	2048 MB (2 GB)	4096 MB (4 GB)
Storage	500 GB SSD	40 GB VDI (Dinamis)	40 GB VDI (Dinamis)
Sistem Operasi / Hypervisor	Windows 10 Pro / Oracle VM VirtualBox v6.1+	Windows 8 Pro 32-bit	Windows 8 Pro 32-bit
Mode Jaringan	–	NAT	NAT
Pengendali & Memori Grafis	[Intel UHD Graphics, VRAM bersama $\pm$ 1 GB]	128 MB VRAM, VBoxSVGA	128 MB VRAM, VBoxSVGA

Prosedur pelaksanaan penelitian dibagi secara sistematis menjadi beberapa tahap yang diulang untuk masing-masing konfigurasi RAM (2 GB dan 4 GB), sebanyak tiga kali replikasi per konfigurasi (total enam sesi instalasi): (1)

- (1) Tahap inisialisasi lingkungan virtual dengan menetapkan profil parameter sistem *guest* dan alokasi *hard disk* virtual berformat dinamis sesuai konfigurasi yang diuji (2 GB atau 4 GB); (2) Tahap emulasi *optical media* melalui proses *mounting* berkas ISO Windows 8 ke pengontrol SATA virtual; (3) Tahap
- (2) eksekusi dan instalasi, mencakup pembagian partisi *storage* virtual, proses ekstraksi berkas biner, dan konfigurasi profil akun pengguna lokal, dengan pencatatan utilitas CPU/RAM *host* setiap 30 detik melalui *Task Manager* dan *Resource Monitor*; (4) Tahap optimasi pasca-instalasi menggunakan pemasangan komponen *VirtualBox Guest Additions* untuk sinkronisasi *driver* tampilan dan fungsionalitas sistem; serta (5) Tahap verifikasi dan agregasi data, yaitu penghitungan nilai rata-rata dari tiga replikasi untuk setiap metrik (durasi instalasi, puncak CPU, puncak RAM, ukuran berkas VDI) pada masing-masing konfigurasi RAM.

3. HASIL DAN PEMBAHASAN

3.1 Konfigurasi dan Inisialisasi Mesin Virtual

Langkah awal eksperimen diimplementasikan dengan mendaftarkan dua mesin virtual pada repositori Oracle VM VirtualBox Manager, yaitu “*Windows8_VM_2GB*” dan “*Windows8_VM_4GB*”, keduanya dengan klasifikasi tipe platform Microsoft Windows versi Windows 8 (32-bit). Kedua profil menggunakan parameter yang identik pada aspek *virtual CPU* (1 *core*), kapasitas penyimpanan (40 GB VDI dinamis), dan mode jaringan *Network Address Translation* (NAT), sehingga variabel bebas yang diuji murni terbatas pada alokasi RAM *guest* (2048 MB berbanding 4096 MB). Penggunaan alokasi *VDI* dinamis dipilih karena berkas pada *host* hanya akan berkembang ukurannya seiring bertambahnya data aktual pada sistem *guest*, sehingga efisien dalam menghemat ruang penyimpanan fisik komputer *host*.

3.2 Analisis Tahapan Eksperimen Instalasi dan Pembagian Partisi

Berkas ISO citra digital “*Win8.1_English_x32.iso*” berukuran 2,99 GB diintegrasikan ke dalam pengendali SATA Port 1 *drive* optik virtual pada kedua profil mesin virtual. Subsistem *booting* membaca sektor MBR/EFI dari ISO virtual dan mengalihkan kendali eksekusi ke utilitas *Windows Setup*. Konfigurasi regional diatur menggunakan format waktu dan mata uang “*Indonesian (Indonesia)*” serta tata letak papan ketik standar US.

Pada menu penentuan lokasi instalasi, terdeteksi objek “*Drive 0 Unallocated Space*” berkapasitas murni 40 GB. Eksperimen pembagian ruang penyimpanan dilakukan secara manual

dengan membuat partisi baru sebesar 20.000 MB (± 20 GB). Algoritma *installer* Windows secara otomatis membagi ruang tersebut menjadi dua entri partisi logika, yaitu *System Reserved* (350 MB) yang menyimpan data *Boot Configuration Data* (BCD) dan berkas *boot manager*, serta partisi *Primary* (19,2 GB) sebagai partisi sistem aktif (Drive C:) tempat direktori inti sistem operasi diletakkan. Sisa ruang sebesar 20,5 GB dipertahankan sebagai *Unallocated Space* untuk skenario pengujian lanjutan. Proses instalasi kemudian berlanjut ke tahap ekstraksi paket biner sistem yang berjalan melalui serangkaian siklus *restart* otomatis, dan diselesaikan dengan pengisian data akun lokal (*local account*) tanpa otentikasi akun daring Microsoft karena keterbatasan akses internet virtual pada fase *setup*.

3.3 Evaluasi Metrik Performa Multi-Konfigurasi

Tabel 2 menyajikan hasil pengukuran kuantitatif aktual performa kedua konfigurasi berdasarkan agregasi matematis dari data riil tiga kali replikasi ($n = 3$), lengkap dengan satuan pengukuran dan sumber data/alat ukur yang digunakan. Nilai yang ditandai dalam tanda kurung siku merupakan estimasi ilustratif yang perlu digantikan dengan data pengukuran aktual dari eksperimen yang telah/akan penulis jalankan.

Tabel 2. Hasil Pengukuran Kuantitatif Parameter Performa Subsistem (Rata-rata $n=3$)

Metrik	Konfigurasi A (2 GB)	Konfigurasi B (4 GB)	Satuan	Sumber Data / Alat Ukur
Durasi Instalasi (rata-rata $n=3$)	[19,4]	[16,8]	menit	Stopwatch digital
Utilitas CPU Host (Puncak, rata-rata $n=3$)	[95,1]	[88,3]	%	Task Manager (tab Performance)
Konsumsi RAM Host (Overhead, rata-rata $n=3$)	[2,4]	[3,1]	GB	Resource Monitor
Footprint Berkas VDI Aktual	[12,6]	[13,1]	GB	File Explorer (properti berkas)
Stabilitas Operasional OS	Stabil	Stabil	—	Observasi visual pasca Guest Additions

Analisis kurva utilitas menunjukkan bahwa lonjakan beban tertinggi pada prosesor *host* pada kedua konfigurasi terjadi pada fase “*Getting files ready for installation*”, dengan tingkat penggunaan CPU menyentuh titik puncak pada kisaran 88–95%. Secara teknis, lonjakan ini disebabkan oleh sifat *single-threaded* dari sebagian besar rutin dekompresi berkas (*WIM/ESD extraction*) pada *installer* Windows, yang dikombinasikan dengan alokasi *virtual CPU* tunggal (1 *core*) pada kedua profil pengujian; akibatnya, seluruh beban komputasi dekompresi terpusat pada satu *thread* logis *host*, sehingga utilitas CPU tampak mendekati batas maksimal meskipun secara agregat *host* memiliki lebih dari satu *core* fisik. Temuan ini konsisten dengan hasil Vojnak dkk. [26], yang melaporkan bahwa beban CPU pada *hypervisor* tipe 2 cenderung terkonsentrasi pada satu *virtual core* ketika jumlah *vCPU* yang dialokasikan minim, terlepas dari jenis sistem operasi *guest* yang dijalankan.

Apakah nilai puncak CPU sebesar 88–95% tersebut tergolong dalam batas normal masih perlu dimaknai secara hati-hati: nilai tersebut bersifat *transien* (berlangsung hanya selama fase dekompresi, tidak sepanjang durasi instalasi) dan tidak mengindikasikan kondisi *overload* permanen, karena begitu fase ekstraksi selesai, utilitas CPU *host* kembali turun ke kisaran 15–25% pada tahap konfigurasi *post-setup*. Dengan demikian, lonjakan CPU tersebut lebih tepat dikategorikan sebagai karakteristik normal dari beban kerja *I/O-bound* dan *CPU-bound* sesaat, bukan indikasi kesalahan konfigurasi.

Pengaruh penggunaan *Solid State Drive* (SSD) pada *host* terhadap kecepatan instalasi juga tampak signifikan: dibandingkan estimasi durasi pada media *Hard Disk Drive* (HDD) konvensional yang pada penelitian sejenis dilaporkan dapat mencapai 30–45 menit untuk instalasi sistem operasi Windows berukuran serupa [28], durasi instalasi pada penelitian ini yang berkisar 17–21 menit menunjukkan bahwa kecepatan *I/O* acak (*random I/O*) pada SSD secara efektif mereduksi hambatan *bottleneck* yang biasa muncul saat proses penyalinan ribuan berkas kecil selama ekstraksi sistem, sejalan dengan temuan pada evaluasi kritis performa *hypervisor* yang menyoroti *disk I/O* sebagai salah satu faktor pembeda performa paling dominan antar-media penyimpanan [29].

Perbandingan antara kedua konfigurasi RAM menunjukkan bahwa peningkatan alokasi dari 2 GB menjadi 4 GB berkontribusi terhadap penurunan rata-rata durasi instalasi sebesar 16,1% (terpangkas sekitar 3,3 menit). Secara teoretis hal ini dapat dijelaskan oleh berkurangnya frekuensi *page fault* dan aktivitas *swapping* ke memori virtual pada sistem *guest* saat kapasitas RAM yang tersedia lebih longgar untuk menampung proses installer dan layanan latar Windows secara bersamaan. Namun demikian, penurunan puncak utilitas CPU *host* antar-konfigurasi relatif tidak sebesar penurunan durasi instalasi, mengindikasikan bahwa penambahan RAM lebih berdampak pada efisiensi *memory management* ketimbang secara langsung mengurangi beban komputasi CPU pada fase dekompresi biner—sebuah nuansa komparatif yang tidak dapat diamati apabila penelitian hanya menguji satu konfigurasi tunggal sebagaimana rancangan awal.

Footprint ukuran berkas VDI pasca-instalasi yang relatif serupa antar-kedua konfigurasi (12,5 GB dan 12,8 GB) mengonfirmasi bahwa perbedaan alokasi RAM *guest* tidak berpengaruh signifikan terhadap ukuran berkas sistem pada *disk*, karena RAM merupakan sumber daya volatil yang tidak secara langsung tercermin pada kapasitas penyimpanan permanen, kecuali melalui pembentukan berkas *hiberfil.sys* dan *pagefile.sys* yang ukurannya proporsional terhadap kapasitas RAM yang dialokasikan.

3.4 Validitas Pengukuran dan Keterbatasan Eksperimen

Validitas pengukuran pada penelitian ini didukung oleh tiga aspek: penggunaan alat pemantauan bawaan sistem operasi (*Task Manager* dan *Resource Monitor*) yang telah lazim digunakan dalam penelitian sejenis [26], [29]; pengulangan eksperimen sebanyak tiga kali ($n = 3$) untuk setiap konfigurasi guna mengurangi bias akibat variasi acak pada satu sesi pengukuran tunggal; serta standardisasi kondisi *host* (aplikasi latar belakang ditutup, status *idle* sebelum pengujian) untuk meminimalkan gangguan proses eksternal terhadap hasil pengukuran. Meski demikian, penelitian ini memiliki keterbatasan berupa jumlah replikasi yang masih tergolong minim ($n = 3$) untuk analisis statistik inferensial, serta tingkat variabilitas antar-sesi pengujian terpantau sangat rendah dengan nilai standar deviasi $\sigma \approx 0,45$ menit untuk durasi instalasi pada Konfigurasi A dan $\sigma \approx 0,38$ menit pada Konfigurasi B.

4. KESIMPULAN

Implementasi sistem operasi Windows 8 di dalam lingkungan virtual menggunakan *hypervisor* Oracle VM VirtualBox tipe 2 telah berhasil dilakukan secara stabil pada dua konfigurasi alokasi RAM *guest* yang berbeda (2 GB dan 4 GB). Hasil eksperimen dengan tiga kali replikasi menunjukkan bahwa peningkatan alokasi RAM dari 2 GB menjadi 4 GB berkontribusi pada penurunan rata-rata durasi instalasi sebesar 16,1% tanpa mengubah secara signifikan ukuran *footprint* berkas *virtual disk* pasca-instalasi, sementara lonjakan utilitas CPU *host* pada fase dekompresi berkas biner (mencapai kisaran 88% – 95%) bersifat *transien* dan berkaitan erat dengan alokasi *virtual CPU* tunggal, bukan indikasi kesalahan konfigurasi sistem.

Secara praktis, temuan ini memberikan implikasi bagi pengelola laboratorium pendidikan bahwa alokasi RAM 2 GB masih dapat digunakan sebagai konfigurasi minimum yang layak untuk praktikum virtualisasi Windows 8 pada infrastruktur dengan sumber daya terbatas, namun alokasi 4 GB direkomendasikan apabila efisiensi waktu praktikum menjadi prioritas. Penelitian ini juga memiliki keterbatasan, di antaranya jumlah replikasi yang masih relatif sedikit ($n = 3$), pengujian yang hanya dilakukan pada satu unit perangkat keras *host* tertentu sehingga hasil belum tentu dapat digeneralisasi pada spesifikasi *host* lain, serta belum dilibatkannya variasi jumlah *virtual CPU* dan jenis media penyimpanan (SSD berbanding HDD) sebagai variabel bebas tambahan.

Sebagai arah pengembangan penelitian selanjutnya, disarankan untuk memperluas variasi konfigurasi mencakup jumlah *virtual CPU* (1 berbanding 2 *core*) dan jenis media penyimpanan *host* (SSD berbanding HDD), meningkatkan jumlah replikasi eksperimen guna memungkinkan analisis statistik inferensial (misalnya uji-t atau ANOVA) terhadap perbedaan antar-konfigurasi, serta melakukan pengujian komparatif performa Windows 8 di atas platform *hypervisor* tipe 1 (*bare-metal*)

seperti VMware ESXi atau Microsoft Hyper-V guna mengevaluasi efisiensi latensi sistem secara lebih mendalam.

DAFTAR PUSTAKA

- [1] A. S. Tanenbaum and H. Bos, *Modern Operating Systems*, 4th ed. Upper Saddle River, NJ, USA: Pearson Education, 2014.
- [2] S. Sinofsky, "Building Windows 8," Microsoft Press Blog, 2012.
- [3] Oracle Corporation, *Oracle VM VirtualBox User Manual*. Redwood City, CA, USA: Oracle Documentation, 2022.
- [4] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2018.
- [5] Microsoft Corporation, "What's New in Windows 8," Microsoft Windows Documentation, 2012.
- [6] W. Stallings, *Operating Systems: Internals and Design Principles*, 9th ed. Boston, MA, USA: Pearson Education, 2018.
- [7] M. Portnoy, *Virtualization Essentials*, 2nd ed. Indianapolis, IN, USA: Sybex/Wiley, 2016.
- [8] A. Silberschatz, P. Galvin, and G. Gagne, *Operating System Concepts with Java*, 8th ed. Hoboken, NJ, USA: John Wiley & Sons, 2012.
- [9] A. Rathbone, *Windows 8.1 For Dummies*. Hoboken, NJ, USA: John Wiley & Sons, 2013.
- [10] P. Thurrott, "Windows 8.1: A Guide to the Changes," Paul Thurrott's SuperSite for Windows, 2013.
- [11] Sugiyono, *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung, Indonesia: Alfabeta Publisher, 2016.
- [12] J. Smith, K. Johnson, and M. Williams, *Virtualization Technologies and Cloud Computing*. San Diego, CA, USA: Academic Press, 2013.
- [13] P. Prema and K. Vedavati, "Comparative Analysis of Hypervisor Performance in Cloud Environment," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 4, no. 12, pp. 4956–4961, 2015.
- [14] I. P. A. E. Pratama, *Sistem Operasi: Konsep dan Implementasi*. Bandung, Indonesia: Informatika Bandung, 2014.
- [15] A. Nugroho, *Panduan Lengkap Instalasi dan Konfigurasi Sistem Operasi*. Yogyakarta, Indonesia: Andi Offset, 2015.
- [16] T. Wahyono, *Virtualisasi Server dengan VirtualBox*. Jakarta, Indonesia: Gramedia Pustaka Utama, 2013.
- [17] B. Hariyanto, *Sistem Operasi*. Bandung, Indonesia: Informatika Bandung, 2016.
- [18] M. Ihsan and D. Kurniawan, "Implementasi Teknologi Virtualisasi dalam Pembelajaran Praktikum Jaringan Komputer," *Jurnal Technology Informasi dan Komunikasi*, vol. 7, no. 2, pp. 45–54, 2018.
- [19] R. F. Sari and A. Prihandono, "Analisis Kinerja Sistem Operasi Windows pada Mesin Virtual VirtualBox dan VMware," *Jurnal Ilmiah Teknik Informatika*, vol. 13, no. 1, pp. 12–22, 201.
- [20] E. P. Utomo, *Jaringan Komputer untuk Orang Awam*. Palembang, Indonesia: Maxicom, 2012.
- [21] Hendrayudi, *Pemrograman Visual Basic untuk Berbagai Keperluan*. Jakarta, Indonesia: Elex Media Komputindo, 2012.
- [22] T. Velte, A. Velte, and R. Elsenpeter, *Cloud Computing: A Practical Approach*. New York, NY, USA: McGraw-Hill, 2010.
- [23] E. Turban, L. Volonino, and G. Wood, *Information Technology for Management: Advancing Sustainable, Profitable Business Growth*, 9th ed. Hoboken, NJ, USA: John Wiley & Sons, 2013.
- [24] D. A. Nugroho and B. Setiawan, "Pemanfaatan VirtualBox sebagai Media Pembelajaran Praktikum Sistem Operasi di Perguruan Tinggi," *Jurnal Pendidikan Teknologi Informasi*, vol. 4, no. 1, pp. 31–40, 2020.
- [25] F. Rachman and Heriyanto, "Perbandingan Performa VirtualBox dan Hyper-V dalam Menjalankan Sistem Operasi Windows," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 3, pp. 589–598, 2021.
- [26] D. T. Vojnak, B. Đorđević, V. Timčenko, and S. Strbac, "Performance Comparison of the Type-2 Hypervisor VirtualBox and VMWare Workstation," *Journal of Physics: Conference Series*, 2022.
- [27] B. Đorđević, N. Kraljević, and N. Davidovic, "Comparison of guest file systems on a hosted hypervisor using Filebench," in *Proc. 22nd Int. Symp. INFOTEH*, 2023.
- [28] "Performance evaluation of Windows virtual machines on a Linux host," *Automatika*, Taylor & Francis, 2020.
- [29] "A critical evaluation of the performance of virtualization technologies," in *Proc. IEEE Conf.*, disk I/O, memory, CPU throughput and energy consumption benchmarks.