

IMPLEMENTASI VIRTUALISASI MENGGUNAKAN XEN HYPERVISOR

Isnan Nofarianto Kurniawan

D3 Manajemen Informatika, Fakultas Teknik, Universitas Negeri Surabaya, isnan.nofarianto.k@gmail.com

Ibnu Febry Kurniawan

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya, ibnufebry@unesa.ac.id

Abstrak

Virtualisasi dalam dunia komputasi memiliki peranan penting. Dengan adanya virtualisasi ini memungkinkan satu *server* bisa diisi beberapa operating sistem. Salah satu aplikasi Hypervisor untuk keperluan ini adalah Xen Hypervisor yang sifatnya open source. Penelitian ini mencoba melakukan penerapan Hypervisor tersebut pada komputer PC desktop pada sistem operasi Debian. Penulis menggunakan tool stack Virt-Manager untuk melakukan konfigurasi. Dari hasil eksperimen didapatkan hasil bahwa Xen dapat mengatur sumber daya CPU, memori (RAM), serta jaringan virtual. Lebih jauh lagi, Xen menggunakan sumber daya lebih hemat pada domU, atau lebih kecil daripada yang dialokasikan pada dom0.

Kata Kunci : *virtual, virtualization, server, xen hypervisor.*

Abstract

Virtualization in the computing world has an important role. With the virtualization allows a single server can be filled multiple operating systems. One application for this purpose Hypervisor is Xen Hypervisor which is open source. This study tried to do the Hypervisor application on a desktop PC computer the Debian operating system. The author uses tool stack Virt-Manager to configure. From the experimental results showed that Xen can adjust the CPU resources, memory (RAM), and virtual network. Furthermore, Xen uses resources more sparingly in domU, or smaller than that allocated to dom0.

Keywords: *virtual, virtualization, server, xen hypervisor.*

PENDAHULUAN

Virtualisasi merupakan strategi untuk mengurangi konsumsi daya pusat data. Dengan virtualisasi, satu host *server* fisik memiliki banyak *server* virtual. Virtualisasi memungkinkan pusat data untuk mengkonsolidasikan infrastruktur *server* fisik dengan menempatkan *virtual* pada sejumlah kecil *server* fisik yang lebih kuat, sehingga menggunakan energy listrik yang sedikit dan menyederhanakan pusat data. Namun akan tetapi permasalahan yang dihadapi penulis adalah virtualisasi sendiri banyak yang tidak gratis, dan spesifikasi yang dibutuhkan untuk membangun virtualisasi sendiri harus bagus. Yang sudah dicapai oleh penulis adalah mendapatkan Hypervisor, yaitu Xen Hypervisor yang sifatnya open source, dan menggunakan Virt-Manager sebagai toolstack.

Tujuan dari penelitian Implementasi Virtualisas Menggunakan Xen Hypervisor adalah dapat membangun dan mengelola virtualisasi dengan menggunakan *Xen Hypervisor* dengan dom0 sebagai pengelola sistem operasi tamu (domU). Dapat mengelola jaringan yang ada pada *Xen Hypervisor* dengan membridge jaringan Xen

dengan *server* Debian. Dengan adanya duplikasi sistem operasi tamu memudahkan untuk mengcopy sebuah domU tanpa harus menginstal lagi dari awal. Alokasi sumber daya memudahkan admin atau pengguna untuk merubah secara dinamis sesuai kebutuhan yang diinginkan.

Dengan adanya virtualisasi dengan dom0 sebagai pengelola sistem operasi tamu (domU) adalah dengan adanya virtualisasi ini dapat menjadi sebagai penghematan biaya pengadaan sebuah *server*. Dengan duplikasi sistem operasi tamu, dapat secara cepat menggandakan domU tanpa harus menginstal mulai dari awal, dan juga sebagai backup *server*. Dengan adanya alokasi sumber daya, dapat dengan mudah merubah dan menambahkan hardisk sesuai dengan yang diharapkan dan menyesuaikan sesuai kebutuhan dengan mudah.

KAJIAN PUSTAKA

Virtualisasi

Di dalam ilmu komputer, istilah virtualisasi (Inggris: *virtualization*) adalah ilmu umum yang mengacu kepada abstraksi dari sumber daya komputer. Dari

definisi lain adalah "sebuah teknik untuk menyembunyikan karakteristik fisik dari sumber daya komputer dan bagaimana sistem lain, aplikasi atau pengguna berinteraksi dengan sumber daya tersebut. Hal ini termasuk membuat sebuah sumber daya tunggal seperti server, sebuah sistem operasi, sebuah aplikasi atau peralatan penyimpanan terlihat berfungsi sebagai beberapa sumber daya logikal, atau dapat juga termasuk definisi untuk membuat beberapa sumber daya fisik (seperti beberapa peralatan penyimpanan atau server) terlihat sebagai satu sumber daya logikal. (Pratt, Ian. 2006).

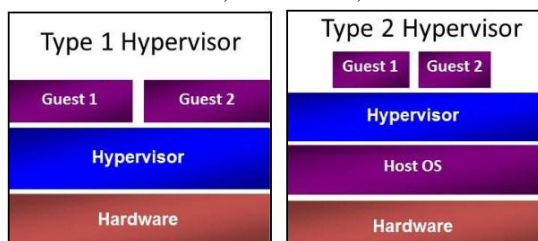
Hypervisor

Hypervisor merupakan suatu landasan bagi virtualisasi atau *software* khusus menyebabkan berbagai sistem operasi dapat berjalan secara simultan pada sebuah komputer. *Hypervisor* kadangkala disebut juga *virtual machine monitor* (VMM).

Banyak yang berpendapat bahwa *hypervisor* merupakan sebuah teknologi baru. Sebenarnya tidak. *Hypervisor* sudah ada sejak tahun 1970 yang diimplementasikan pada IBM CP-370 yang merupakan hasil reimplementasi dari CP-67 untuk system/370. *Hypervisor* disebut juga VM/370. VM/370 sekarang dikenal dengan z/VM yang menjadi dasar bagi virtualisasi Linux dan OpenSolaris pada IBM System z.

Secara umum ada dua jenis *hypervisor*, jenis pertama atau Type 1 dan kedua atau Type 2.

1. *Hypervisor* Type 1 (*Bare-Metal Architecture*) berjalan secara langsung pada hardware yang digunakan. Artinya tidak diperlukan sistem operasi untuk menjalankan *hypervisor*, *Hypervisor* dapat diinstal langsung pada disk dan tidak memerlukan sistem operasi Windows, Linux, MacOS, dan sebagainya. Contoh *hypervisor* jenis ini adalah Citrix XenServer, Xen Cloud Platform, Vmware ESX/ESXi, Microsoft Hyper-V.
2. *Hypervisor* Type 2 (*Hosted Architecture*) merupakan sebuah aplikasi yang diinstal di atas sistem operasi yang umum. menginstal sistem operasi seperti Windows, Linux, MacOS dan sebagainya kemudian menginstal *hypervisor* di atas sistem operasi tersebut. Contoh *hypervisor* tipe ini adalah Vmware GSX Server/Workstation, VirtualBox, VirtualPC.



Gambar 1. Type 1 dan Type 2 Hypervisor

Dari segi performa, Type 1 akan bekerja lebih baik dibandingkan Type 2. *Hypervisor* dapat berjalan langsung di atas hardware, sehingga resources komputer yang tersedia dapat dimanfaatkan secara optimal oleh *hypervisor*. Jika stabilitas dan kecepatan menjadi alasan utama maka Type 1 adalah pilihan yang tepat (Iwan Sofana, 2012).

Xen Hypervisor

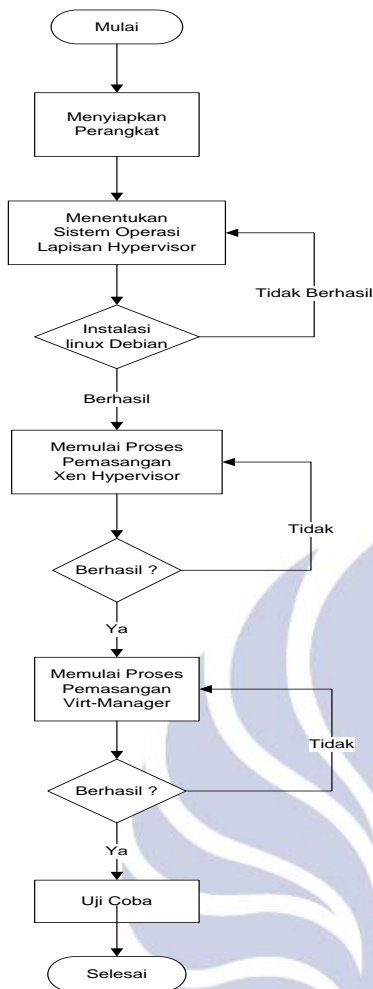
Xen Hypervisor adalah layer perangkat lunak yang berjalan secara langsung pada perangkat keras yang menggantikan sistem operasi sehingga memungkinkan perangkat keras komputer tersebut untuk menjalankan beberapa *guest* sistem operasi secara bersamaan. Beberapa processor yang mendukung antara lain x86-64, Itanium, Power PC, dan processor ARM membuat Xen Hypervisor memungkinkan untuk berjalan di berbagai perangkat computer dan saat ini didukung oleh Linux, NetBSD, FreeBSD, Solaris, Windows dan sistem operasi umum lainnya yang sebagai *Sistem operasi tamu* pada Hypervisor. Komunitas Xen.org mengembangkan dan memelihara Xen Hypervisor sebagai solusi gratis yang berlisensi dibawah GNU General Public License

Virt-Manager

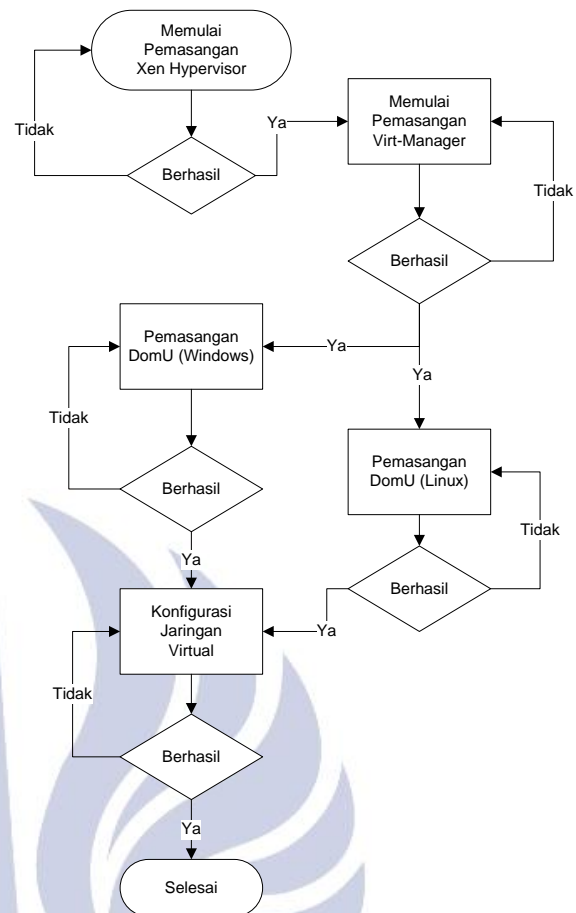
Virtual Machine Manager atau disingkat Virt-Manager adalah suatu aplikasi dekstop pengelola virtual machine dengan berbasis tampilan GUI. pada aplikasi ini juga dapat memonitoring penggunaan CPU, RAM, I/O dan jaringan secara real-time dalam bentuk grafik sehingga mempermudah dalam pembuatan virtual machine baru dan monitoring sistem pada virtual machine serta melakukan konfigurasi pada sistem. Aplikasi ini dibuat dengan bahasa pemrograman python dan UI dengan Glade dan Gtk+ dimana dibuat oleh libvirt. alasan penggunaan aplikasi ini adalah kontribusi libvirt yang dalam pembuatan aplikasi monitoring virtual machine diawali pada *hypervisor* Xen sehingga lebih kompeten dalam aplikasi monitoring virtual machine.

METODE

Penelitian dimulai dengan tahapan melakukan analisis kebutuhan, serta pemasangan. Tahapan analisis kebutuhan dapat dilihat pada gambar 2. Pada gambar tersebut terlihat bahwa proses diawali menentukan sistem operasi lapisan *hypervisor*, lalu memulai proses pemasangan Xen Hypervisor dan memulai proses pemasangan Virt-Manager.

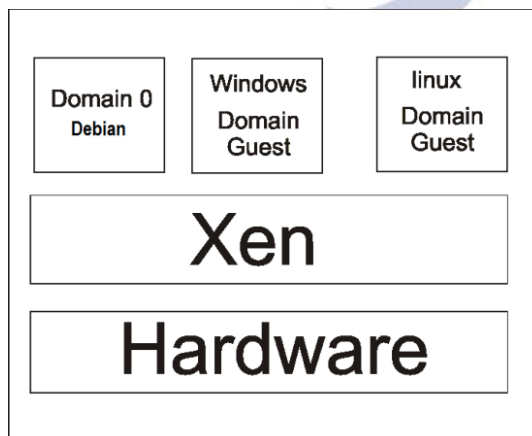


Gambar 2. Diagram Alir Analisis Kebutuhan



Gambar 4. Diagram Alir Analisis Pemasangan Xen Hypervisor dan Virt-Manager

Selanjutnya, desain dari sistem yang akan berjalan terlihat seperti pada gambar 3.



Gambar 3. Desain Sistem

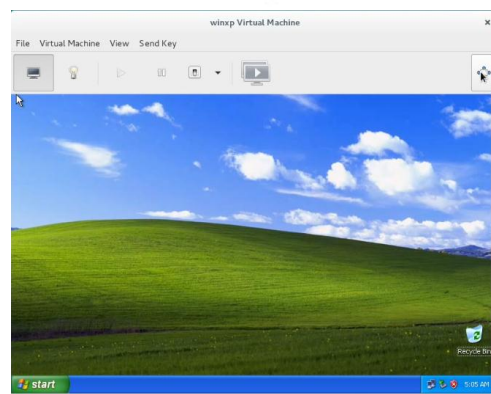
Pemasangan dan pengelolaan melalui beberapa tahapan. Tahapan diantaranya, memulai pemasangan Xen Hypervisor dan Virt-Manager, kemudian pemasangan domU lalu konfigurasi jaringan virtual.

HASIL DAN PEMBAHASAN

Hasil Pengujian DomU

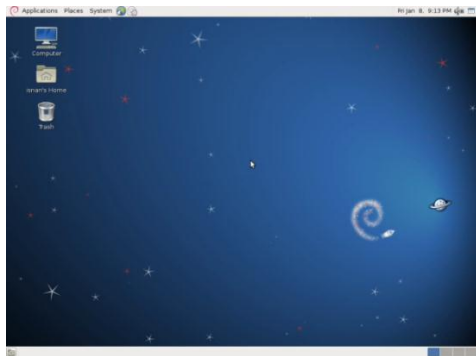
Dilakukan pengujian untuk mengetahui apakah domU dapat berhasil dibuat dan berjalan dengan dikontrol oleh dom0, pada pengujian domU dilakukan dua percobaan menggunakan OS Windows XP dan Linux Debian dan hasilnya berhasil seperti pada gambar 5 dan gambar 6

Pada gambar 5 terlihat tampilan desktop dari domU Windows XP.



Gambar 5. Pengujian DomU Windows XP

Pada gambar 6 terlihat tampilan desktop dari domU Linux Debian.

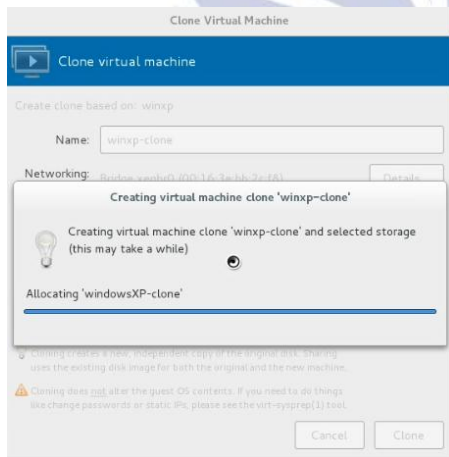


Gambar 6. Pengujian DomU Linux Debian

Hasil Pengujian Duplikasi

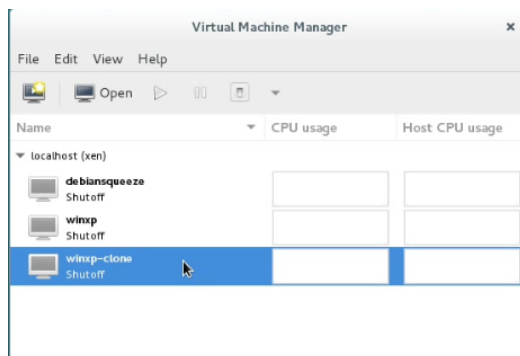
Dilakukan pengujian untuk mengetahui apakah duplikasi dapat dilakukan pada kedua domU yang telah dibuat pada pengujian sebelumnya, dan hasilnya berhasil seperti yang terlihat pada gambar 7, gambar 8, gambar 9 Dan gambar 10.

Proses dari duplikasi Windows XP, proses duplikasi berlangsung kurang lebih 15 menit.



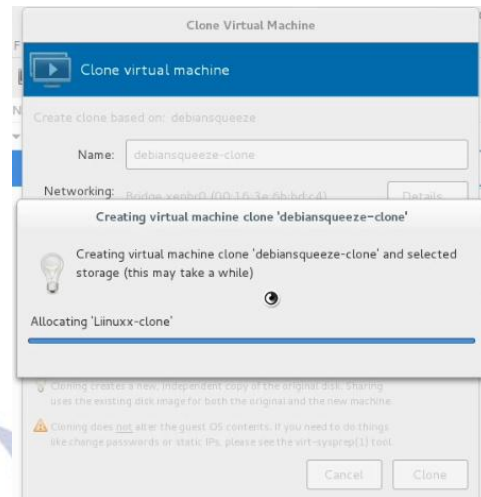
Gambar 7. Proses Duplikasi Windows XP

Tampilan domU Windows XP yang telah berhasil diduplikasi dengan nama winxp-clone.



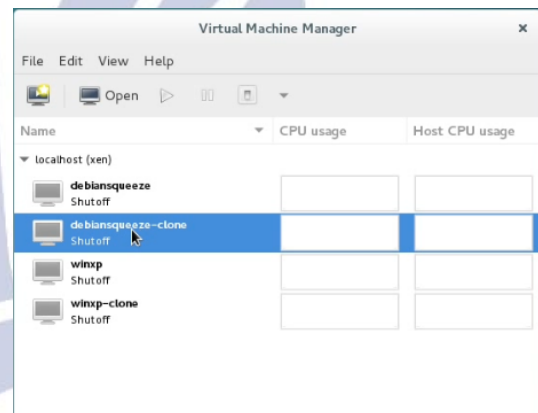
Gambar 8. Pengujian Duplikasi Windows XP

Proses dari duplikasi Linux Debian, proses duplikasi berlangsung kurang lebih 20 menit.



Gambar 9. Proses Duplikasi Linux Debian

Tampilan domU Linux Debian yang telah berhasil diduplikasi dengan nama debiansqueeze-clone.



Gambar 10. Pengujian Duplikasi Linux Debian

Hasil Pengujian Alokasi Sumber Daya

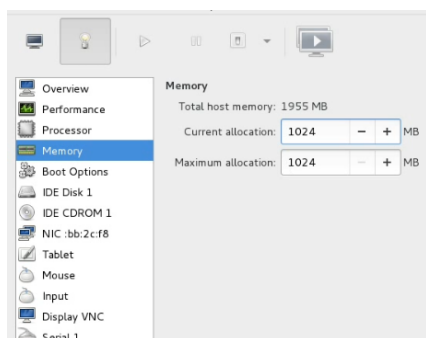
Dilakukan pengujian untuk mengetahui apakah pengujian alokasi sumber daya dapat dilakukan pada kedua domU yang sebelumnya telah dibuat, untuk pengujian ini akan dilakukan penambahan RAM dan juga hardisk pada domU Windows XP dan Linux Debian, dan hasil pengujian seperti pada dibawah ini.

Tampilan awal dari RAM domU Windows XP.



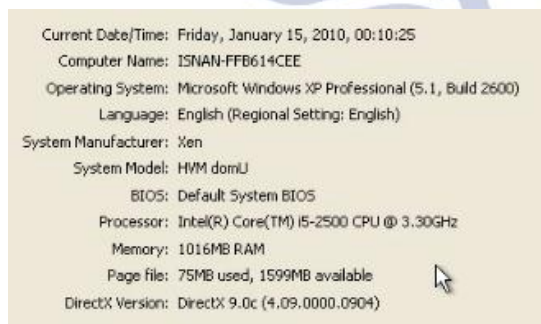
Gambar 11. Tampilan Awal Alokasi RAM DomU Windows XP

Kemudian rubah alokasi RAM sebesar 1024 MB.



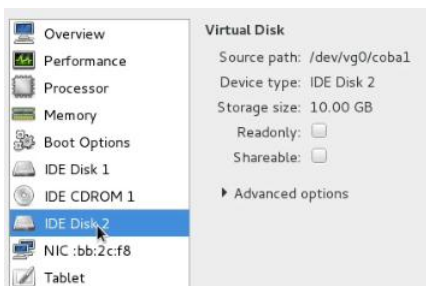
Gambar 12. Alokasi RAM DomU Windows XP

Tampilan sesudah RAM domU dirubah menjadi sebesar 1024 MB dan seperti pada gambar.14 pengujian telah berhasil dilakukan.



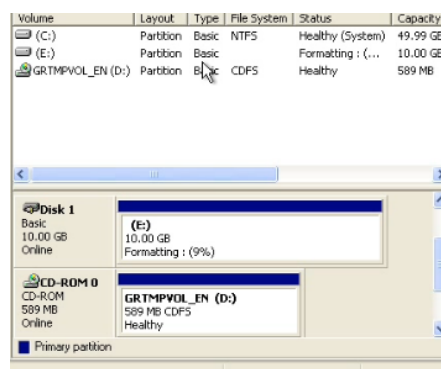
Gambar 13. Tampilan Setelah Alokasi RAM

Proses penambahan virtual hardisk pada domU Windows XP seperti pada gambar.15.



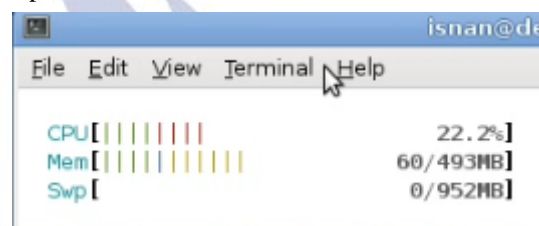
Gambar 14. Penambahan Virtual Hardisk DomU Windows XP

Terlihat pada gambar.16 virtual hardisk (E) sebesar 10 GB



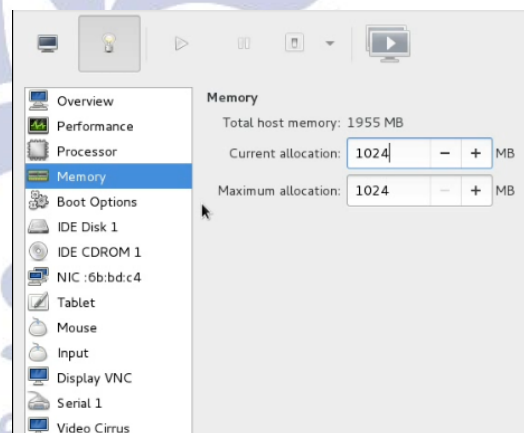
Gambar 15. Tampilan Disk Manager

Tampilan awal dari RAM domU Linux Debian.



Gambar 16. Tampilan Awal Alokasi RAM DomU Linux Debian

Kemudian rubah alokasi RAM sebesar 1024 MB.



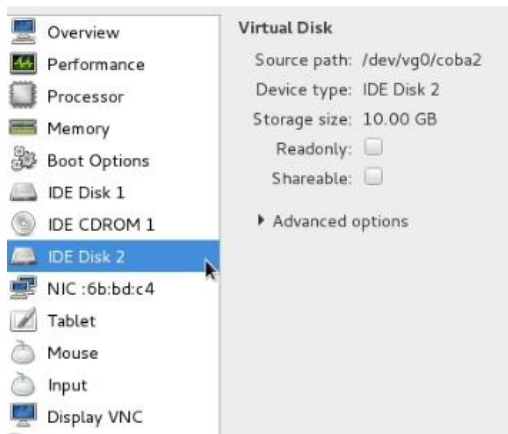
Gambar 17. Alokasi RAM DomU Linux Debian

Tampilan sesudah RAM domU dirubah menjadi sebesar 1024 MB dan seperti pada gambar.18 pengujian telah berhasil dilakukan.



Gambar 18. Tampilan Setelah Alokasi RAM Debian

Proses penambahan virtual hardisk pada domU Linux Debian seperti pada gambar.19.



Gambar 19. Penambahan Virtual Hardisk DomU Linux Debian

Terlihat pada gambar.20 pada /dev/sda2 telah tertambahkan virtual hardisk sebesar 10 GB.

```
root@debian:/home/isanan# fdisk -l
```

```
Disk /dev/sda: 53.7 GB, 53687091200 bytes
255 heads, 63 sectors/track, 6527 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0007c620
```

```
Device Boot      Start         End      Blocks   Id  System
/dev/sda1        1         4098     32909312   83  Linux
/dev/sda2       4098        4219       975873    5  Extended
/dev/sda5       4098        4219       975872    82  Linux swap / Solaris
```

Gambar 20. Telah Tertambahkan Virtual Hardisk Dari DomU Linux Debian

Hasil Pengujian Resource

Dilakukan pengujian ini bertujuan untuk membandingkan antara realisasi dan alokasi penggunaan sumber daya memori pada dom0 dan domU. Pada tiap skenario nilai RAM yang terpakai pada dom0 diambil dari mengeksekusi perintah "xltop", sedangkan pada domU menggunakan perintah top dan task manager. Terdapat 2 jenis Operating Sistem mesin virtual yang dipakai, yakni Windows XP dan Linux Debian.

Pengujian ini bertujuan untuk membandingkan realisasi penggunaan memori oleh dom0 dan penggunaan memori domU. Skenario akan membandingkan beberapa konfigurasi alokasi RAM sebesar 256 MB, 512 MB, 768 MB, dan 1024 MB. Hasil pengujian dapat dilihat pada tabel 1 dan 2.

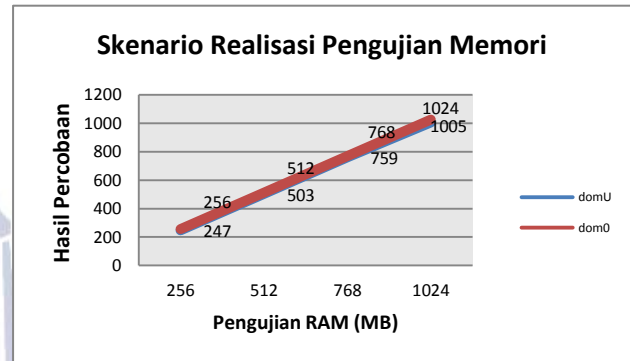
Tabel.1. Tabel Pengujian Dom0 Windows XP

Pengujian Realisasi Penggunaan Memori				
Dom0	256	512	768	1024
	256	512	768	1014

Tabel.2. Tabel Pengujian DomU Windows XP

Pengujian Realisasi Penggunaan Memori				
DomU	256	512	768	1024
	247	503	759	1005

Berdasarkan tabel 1 dan 2, dapat dibuat grafik seperti pada gambar 21.



Gambar 21. Grafik Windows XP

Pengujian ini bertujuan untuk membandingkan realisasi penggunaan memori oleh dom0 dan penggunaan memori domU. Skenario akan membandingkan beberapa konfigurasi alokasi RAM sebesar 128 MB, 256 MB, 512 MB, 768 MB, dan 1024 MB. Hasil pengujian dapat dilihat pada tabel 3 dan 4.

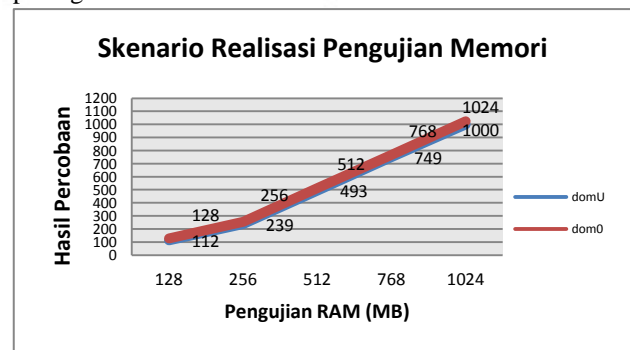
Tabel 3. Tabel Pengujian Dom0 Linux Debian

Pengujian Realisasi Penggunaan Memori					
Dom0	128	256	512	768	1024
	128	256	512	768	1024

Tabel 4. Tabel Pengujian DomU Linux Debian

Pengujian Realisasi Penggunaan Memori					
Dom0	128	256	512	768	1024
	112	239	493	749	1000

Berdasarkan tabel 3 dan 4, dapat dibuat grafik seperti pada gambar 22.



Gambar 22. Grafik Linux Debian

PENUTUP

Kesimpulan

Berdasarkan pembahasan-pembahasan pada bab sebelumnya dapat disimpulkan bahwa

1. Hypervisor Xen dapat bekerja pada server Debian Jessie dan dapat berjalan stabil serta berjalan normal, pengujian domU pun dapat berjalan pada bantuan toolstack virt-manager.
2. Kloning sistem operasi tamu (guest) Debian dan Windows XP pun dapat dilakukan dan hasil dari kloning pun dapat berjalan normal serta jaringan pada domU kloning pun dapat tersambung dan berjalan dengan lancar.
3. Alokasi sumber daya pada dua sistem operasi tamu yang dilakukan memudahkan admin server untuk mengelola server dikarenakan jika dibutuhkan RAM serta hardisk admin server tinggal merubah serta menambahkan sesuai keinginan.

Saran

Diharapkan dapat sebagai bahan referensi dan sebagai bahan tambahan masukan pengetahuan, pemahaman kepada pembaca mengenai Xen Hypervisor dengan toolstack virt-manager. Diharapkan pada penelitian selanjutnya juga mampu melakukan penelitian dengan menggunakan metode toolstack dan dengan server yang berbeda.

DAFTAR PUSTAKA

Ashari, Ahmad, dkk. 2010. *Linux Sistem Administrasi*. Bandung: Informatika.

Bowo, Adi. 2016. <http://adibowo.com/komputer-server-pengertian-jenis-dan-fungsi-server/>

Chisnall, David. 2008. *The Definitive Guide to the Xen Hypervisor*. San Fransisco: Prentice Hall

Farisy, Achmad. 2012. *Implementasi Dan Analisa Kinerja Xen Virtual Server*. Jakarta : Ui

Kristiyanto, Ari. 2011. *Virtualisasi Menggunakan Xen Hypervisor Sebagai Efisiensi Penggunaan Server*. Jakarta : Universitas Islam Negeri Syarif Hidayatullah

Matthews, N Jeanna. 2008. *Running Xen*. U.S. Prentice Hall

Pratt, Ian. 2006. *Xen and the Art of Virtualization*.

P. O'Rourke a, M. Keefe. 2001. *Performance Evaluation of Linux Virtual Server*.

Takemura, Chris, Crawford. 2009. *The Book of XEN*. San Fransisco: No Starch Press.