**BAB I**

**PENDAHULUAN**

**1.1. Latar Belakang**

Fungsi internet yang pada awalnya hanya sebagai media pengirim data dan informasi, pada era ini mengalami perluasan menjadi semakin berkembang mengikuti bidangnya masing-masing. Sebagai contoh internet digunakan untuk mengelola data pribadi dan juga pekerjaan penting seperti pekerjaan kantor. Dengan demikian internet diharapkan dapat memberikan layanan berupa kemudahan serta kenyamanan dalam melakukan aktifitas kegiatan sehari-hari. Perpaduan permasalahan tersebut yang kemudian mengarah kepada sebuah teknologi, yaitu komputasi berbasis internet. Dalam perkembangannya, teknologi komputasi yang berbasis internet lebih diarahkan kepada proses pengaplikasian sistem yang mudah dan tidak perlu banyak waktu dan tenaga. Teknologi yang sudah ada hingga saat ini masih sangat terbatasi oleh ruang, yang tentunya pasti akan memerlukan banyak waktu dan tenaga.

Contoh di bidang pendidikan yaitu instansi pada perguruan tinggi. Kebutuhan pengolahan data yang ada sangat tinggi. Data-data tersebut merupakan data yang diolah oleh berbagai divisi dalam instansi tersebut. Data yang diolah dan disimpan pada sistem tersebut pasti semakin lama akan semakin bertambah, sehingga perlu tempat penyimpanan atau storage yang besar pula. Penambahan kapasitas ini juga mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya menjadi semakin bertambah. *Bandwidth* yang diperlukan untuk proses ini pun tidak sedikit.

Sejalan dengan hal di atas, Pada Universitas Bina Darma juga telah memiliki banyak sekali sistem informasi yang digunakan untuk mendukung aktifitasnya. Salah satunya adalah sistem informasi Pasca Sarjana Universitas Bina Darma. Server konvensional akan di batasi oleh jumlah *core processor*, *harddisk* dan *memory*. Dengan keterbatasan fisik yang ada maka kita tidak mungkin membebani sebuah server konvensional dengan beban maksimal. Jika *resource* / sumber daya habis, maka biasanya kita harus menginstall ulang seluruh aplikasi dan data di server yang kapasitasnya lebih besar dan memigrasi semua aplikasi yang ada ke server yang baru. Ini akan membutuhkan waktu 1-2 hari untuk menyiapkan sebuah server baru, itupun kalau tidak ada masalah. Data pada sistem tersebut semakin lama akan semakin bertambah dan itu membuat diperlukannya tempat penyimpanan atau *storage* yang besar. Dengan semakin bertambahnya data-data pada sistem tersebut dapat pula mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya semakin bertambah.

Oleh karena itu, salah satu solusi permasalahan tersebut yaitu menyediakan tempat penyimpanan atau storage yang besar, yang tidak mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya menjadi bertambah.*cloud computing* ditunjuk sebagai teknologi yang dapat mengatasi masalah tersebut. Teknologi ini menggabungkan prinsip dasar ekonomi dan peletakan sumber daya komputasi. Sesuai dengan karakteristiknya yaitu virtualisasi sumber daya komputasi dan penyewaan berbasis pengguna. Dalam hal ini penggunaannya pada sistem informasi Pasca Sarjana di Universitas Bina Darma dapat menggunakan *private cloud computing* menggunakan *eucalyptus-ubuntu enterprise cloud.* Yang mana merupakan pemodelan *cloud computing* yang memberikan lingkup yang lebih kecil untuk dapat memberikan layanan kepada pengguna tertentu

* 1. **. Perumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan, maka penulis merumuskan permasalah dalam penelitian ini yaitu “Bagaimana menggunakan *private cloud computing* menggunakan *eucalyptus-ubuntu enterprise cloud* sebagai sistem *cloud computing* yang digunakan pada sistem informasi Pasca Sarjana Universitas Bina Darma?”.

* 1. **. Batasan Masalah**

Dalam penelitian ini penulis membatasi permasalahan agar tetap terarah dan tidak menyimpang dari apa yang sudah direncanakan sebelumnya. pada penelitian ini penulis membatasi masalah pada implementasi sistem *Cloud Computing* menggunakan *Private Cloud Computing* saja.

* 1. **. Tujuan dan Manfaat Penelitian**

**1.4.1 Tujuan Penelitian**

Penelitian ini bertujuan untuk menganalisis penggunaan *private cloud computing* menggunakan *eucalyptus-ubuntu enterprise cloud* pada sistem informasi Pasca Sarjana di Universitas Binadarma agar dapat menyediakan tempat penyimpanan atau storage yang besar, yang tidak mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya menjadi bertambah.

* + 1. **Manfaat Penelitian**

Adapun manfaat penelitian ini adalah :

a. Dapat memahami konsep penerapan *private cloud computing* menggunakan *eucalyptus-ubuntu enterprise cloud* dengan baik guna kenyamanan administrator sistem informasi Pasca Sarjana di Universitas Bina Darma

b. Memahami kelebihan dan kekurangan dari sistem yang akan di uji coba diterapkan di Universitas Bina Darma ini

c. Bagi penulis dapat menambah pengetahuan dan pemahaman tentang sistem yang baik bagi pengguna.

d. Diharapkan bagi pengguna sistem ini nantinya akan merasa lebih terbantu dalam penerapan sistem yang tersedia tersebut.

**1.5. Waktu dan Tempat Penelitian**

**1.5.1. Waktu Penelitian**

Penelitian ini dilaksanakan dari bulan Desember 2012 sampai dengan bulan Juli 2013

* + 1. **Tempat Penelitian**

lokasi penelitian yaitu pada UPT Universitas Bina Darma.

* 1. **Alat dan Bahan**

Adapun alat dan bahan yang digunakan dalam proses penelitian ini meliputi *hardware* dan *software* yang diantaranya sebagai berikut:

1. Perangkat Keras (*Hardware*)
2. Menggunakan 3 buah *PC* masing – masing untuk 2 *server* dan 1 *client* dengan spesifikasi :

* *Intel Core 2 Duo processor T 6600, HD 250 GB, 2GB RAM* untuk server 1 dan Client
* [*Core I5 3470 3.2 GHZ LGA 1155 BOX*](http://www.tokotops.com/product/30/2323/CORE-I5-3470-3-2-GHZ-LGA-1155-BOX/?o=a)*, HD 250 GB, 4GB RAM* untuk server II

1. *Kabel UTP & Konektor RJ 45*
2. *Keyboard* dan *Mouse*
3. *Core Switch Cisco 3750*
4. *Cisco Router 3800*
5. Pernagkat Lunak (*Software)*

Implementasi *private cloud computing* pada Tugas Akhir ini menggunakan sistem operasi *Ubuntu 10.04 Server* versi *UEC* serta perangkat lunak *Eucalyptus* untuk server dan sistem operasi Ubuntu 10.04 untuk *client.*

* 1. **. Metodologi Penelitian**

**1.7.1 Metode Penelitian**

Dalam penelitian ini metode penelitian yang digunakan adalah metode penelitian eksperimen, menurut (Sumadi, 2004) metode eksperimen adalah untuk menyelidiki kemungkinan saling hubungan sebab-akibat dengan cara mengenakan kepada satu atau lebih kelompok eksperimental satu atau lebih kondisi perlakuan dan memperbandingkan hasilnya dengan satu atau lebih kelompok kontrol yang tidak dikenai kondisi perlakuan.

Dengan mengacu pada model penelitian ini penulis melakukan pendekatan dalam langkah-langkah pokok kegiatan penelitian yaitu :

1. Melakukan kajian secara induktif yang berkaitan erat dengan permasalahan yang hendak dipecahkan. Mengidentifikasi komponen-komponen apa saja yang digunakan dalam penelitian dan menentukan objek yang diteliti. Pada langkah pertama ini penulis mengkaji sistem yang sedang berjalan di sistem informasi Pasca Sarjana Universitas Binadarma.

2. Mengidentifikasi, mendefinisikan, menganalisis, mempelajari dan membatasi permasalahan yang akan dibahas dalam penelitian. Ada beberapa pokok permasalahan yang akan dibahas yaitu, seberapa efektif dan efisien sistem yang sedang berjalan sekarang, sekaligus membahas kelemahan atau kekurangan sistem yang sedang berjalan.

3. Studi literatur, mengumpulkan literatur tentang sistem yang sedang berjalan sekarang meliputi spesifikasi *hardware* yang tersedia dalam sistem yang ada.

4. Membuat rencana kegiatan penelitian, pada tahap ini melakukan rencana kegiatan uji coba penerapan *Private Cloud Computing*

5. Melakukan eksperimen, pada tahap ini melakukan uji coba penerapan *Private Cloud Computing* menggunakan *Eucalyptus Ubuntu Enterprise Cloud*. Meliputi penjelasan tools yang digunakan, kegunaan dari tools tersebut,installasi dan penerapan.

6. Mengumpulkan data proses dari eksperimen, selama proses uji coba penerapan sistem *Private* *Cloud Computing* menggunakan *Eucalyptus Ubuntu Enterprise Cloud* pada sistem informasi pasca sarjana Universitas Binadarma. semua data dikumpulkan untuk dianalisis.

7. Mengorganisasikan, mendeskripsikan dan mengelompokan data hasil penerapan sistem *Private Cloud Computing* menggunakan *Eucalyptus Ubuntu Enterprise Cloud* pada sistem informasi pasca sarjana Universitas Binadarma.

8. Menganalisis data dan melakukan tes signifikansi dengan teknik statistika yang relavan untuk menentukan tahap signifikasi hasilnya.

9. Membuat Kesimpulan, menginterpresentasikan hasil dari eksperimen dan membuat kesimpulan dari penerapan sistem *Private Cloud Computing* menggunakan *Eucaliptus Ubuntu Enterprise Cloud* pada sistem informasi Pasca Sarjana Universitas Binadarma.

**1.7.2 Metode Pengumpulan Data**

Dalam hal pengumpulan data, penulis menggunakan beberapa cara yaitu :

* + 1. Studi kepustakaan (*Literature*)

Yaitu data yang didapat melalui *literature*, melakukan studi kepustakaan dalam mencari bahan dari *internet* dan membaca buku yang sesuai dengan objek yang diteliti.

1. Penelitian (*Observasi*)

Data yang diperlukan dikumpulkan dengan melihat secara langsung dari objek yang diteliti pada Server Sitem Informasi Pasca SarjanaUniversitas Bina Darma.

1. Wawancara (*Interview*)

Untuk mendapatkan data-data secara langsung dari sumber yang mengerti sehubungan dengan pengamatan yang penulis lakukan. Dalam hal ini penulis mengajukan pertanyaan-pertanyaan kepada penanggung jawab *server* komputer yaitu kepala bagian administrator UPT-SIM .

**1.8. Sistematika Pembahasan**

Dalam sistematika pembahasan ini akan menjelaskan mengenai uraian secara singkat isi tiap-tiap bab dalam penelitian ini adalah sebagai berikut :

**BAB I PENDAHULUAN**

Pada Bab ini menguraikan latar belakang masalah, tujuan dan manfaat pengamatan, rumusan masalah, pembatasan masalah dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Pada Bab ini menguraikan pengertian mengenai landasan pemikiran yang berisi teori-teori mengenai sistem *Cloud Computing* menggunakan *Private Cloud Computing*

**BAB III TINJAUAN OBJEK**

Pada bab ini menguraikan tinjauan objek penelitian yang berisi sejarah dari sebuah objek yang diteliti, visi dan misi, sistem yang sedang berjalan dan permasalahan yang muncul.

**BAB IV METODOLGI PENELITIAN**

Pada Bab ini menguraikan rancangan yang akan dibuat untuk menyelesaikan permasalahan yang ada.

**BAB V HASIL DAN PEMBAHASAN**

Pada bab ini menguraikan tentang hasil dari implementasi pada bab sebelumnya seperti membahas kelebihan sistem yang digunakan serta kekurangannya.

**BAB VI SIMPULAN DAN SARAN**

Pada Bab ini menguraikan kesimpulan - kesimpulan dari pembahasan bab-bab di atas dan kemudian dilanjutkan dengan saran - saran.

**BAB II**

**TINJAUAN PUSTAKA**

**2.1. Sistem Informasi**

Sistem informasi yaitu suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan.

Sistem informasi di suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

(http://agungsr.staff.gunadarma.ac.id/Downloads/files/3412/Konsep+SI.pdf diakses 05/12/2012)

**2.2. *Cloud Computing***

**2.2.1. Definisi Cloud Computing**

Istilah *cloud* telah banyak digunakan dalam perkembangan dunia Internet, karena Internet bisa dikatakan sebagai sebuah awan besar. Penggunaan istilah ini awalnya digunakan untuk gambaran sebuah jaringan besar/*backbone* yang berupa awan, yang sebenarnya berisi kumpulan komputer yang saling terhubung. Konsep ini dikenalkan pada 1961, ketika Profesor John McCarthy menyatakan bahwa “teknologi komputer *time-sharing* mungkin akan berkembang di masa depan, dimana kemampuan komputasi dan aplikasi spesifik dapat dijual melalui model bisnis berdasarkan tipe utilitas”. Baru pada sekitaran tahun 2008 awal teknologi ini muncul kembali dan masih banyak perbedaan pendapat serta pemahaman mengenai teknologi *cloud computing*. Semua menjadi sangat jelas saat ini sehingga memaksa para profesional IT untuk dapat beradaptasi dengan cepat dalam mengimplementasikan sistem *cloud computing*. *Cloud computing* sendiri adalah sebuah model komputasi, dimana sumber dayanya seperti processor / computing power, storage, network, dan software menjadi abstrak dan diberikan sebagai layanan di jaringan / internet menggunakan pola akses remote. Model billing dari layanan ini umumnya mirip dengan modem layanan publik. Ketersediaan on-demand sesuai kebutuhkan, mudah untuk di kontrol, dinamik dan skalabilitas yang hampir tanpa limit adalah beberapa atribut penting dari cloudcomputing. (Johnson. D.dkk, 2010:1)

Jadi, *Cloud computing* juga dapat dikatakan sebuah “mekanisme” yang memungkinkan pengguna “menyewa” sumber daya teknologi informasi melalui internet dan memanfaatkan dan membayar sesuai dengan layanan yang digunakan saja. Dengan konsep ini, maka semakin banyak orang yang bisa memiliki akses dan memanfaatkan sumber daya tersebut, tanpa harus melakukan investasi besar-besaran.

**2.2.2. Karaketristik Sistem *Cloud Computing***

Server konvensional akan di batasi oleh jumlah core processor, harddisk dan memory. Dengan keterbatasan fisik yang ada maka kita tidak mungkin membebani sebuah server konvensional dengan beban maksimal. Jika resource / sumber daya habis, maka biasanya kita harus menginstall ulang seluruh aplikasi dan data di server yang kapasitasnya lebih besar dan memigrasi semua aplikasi yang ada ke server yang baru. Ini akan membutuhkan waktu 1-2 hari untuk menyiapkan sebuah server baru, itupun kalau tidak ada masalah.

Yang menarik dari Cloud Computing berbeda dengan server konvensional terutama:

• Secara fisik berupa kumpulan hardware / server yang tersambung dalam sebuah jaringan (LAN / WAN). Tetapi dari sisi, pengguna dapat melihat sebagai sebuah komputer besar.

• Idealnya tidak ada batasan dengan kapasitas processor, kapasitas harddisk dan kapasitas memory.

• Idealnya tidak ada batasan dengan berapa jumlah "hosting" server yang berjalan di belakangnya.

• Menambahkan sebuah "hosting" hanya membutuhkan waktu beberapa menit saja.

• Jika ada kekurangan resource (sumber daya), baik itu processor, harddisk maupun memory, kita dapat dengan mudah sekali menambahkan server tambahan dan langsung dapat berintegrasi ke jaringan cloud. Butuh waktu sekitar 20 menit-an untuk menyiapkan server kosong / baru untuk dapat berintegrasi ke jaringan cloud. (Ono W. Purbo, 2012:2)

**2.2.3. Model *Cloud Computing***

**2.2.3.1. Model *Cloud Computing* Berdasarkan Infrastruktur**

Ada beberapa model penyebaran dari *cloud computing* yang dapat diterima oleh para stakeholder saat ini dan diakui oleh *National Institute of Standards and Technology* (NIST) (David E.Y Sarana, 2010) :

***1. Public Cloud***

*Public cloud* merupakan sebuah model layanan *cloud* yang disediakan oleh *provider* dan ditujukan untuk layanan *public*/masal. Mekanisme *public cloud* adalah sebuah utilitas berbasis bayar yang disesuaikan dengan penggunaan. *Resource* dari *cloud* ini dihosting di tempat penyedia layanan, mulai dari aplikasi hingga media penyimpanan/*storage*. Contoh dari *public cloud* yang populer adalah *Amazon AWS* (EC2, S3 dll), *Rackspace Cloud Suite*, dan *Microsoft’s Azure Service Platform*.

***2. Private Cloud***

*Private cloud* dibangun, dioperasikan, dan dikelola oleh sebuah organisasi untuk perusahaan penggunaan/keperluan internal untuk mendukung operasi bisnisnya secara eksklusif. Mulai dari masyarakat umum, perusahaan swasta, hingga organisasi pemerintah di seluruh dunia yang mengadopsi model ini untuk mengeksploitasi manfaat *cloud* seperti fleksibilitas, pengurangan biaya, kecepatan dan sebagainya.

***3. Community Cloud***

*Community cloud* terbagi menjadi beberapa organisasi dan mendukung komunitas tertentu yang telah berbagi kepentingan misalnya misi, persyaratan keamanan, kebijakan, dan pertimbangan. *Community cloud* dikelola oleh sebuah organisasi atau pihak ketiga dan mungkin oleh anggota aktif. Salah satu contoh dari *Community Cloud* adalah *OpenCirrus*, yang dibentuk oleh *HP*, *Intel*, *Yahoo*, dan lainnya.

***4. Hybrid Cloud***

*Hybrid cloud* merupakan infrastruktur yang terdiri dari dua atau lebih *cloud* (*private*, *community*, atau *public*). Jadi, *Hybrid cloud* adalah infrastruktur *cloud* berupa gabungan dari beberapa *cloud* yang ada.

**2.2.3.2. Model *Cloud* Berdasarkan Jenis Layanan**

Layanan ini secara umum dibagi menjadi tiga kategori menurut definisi NIST (David E.Y Sarana, 2010):

**1. *Infrastructure as a Service* (IaaS)**

Sistem memberikan layanan kepada konsumen berupa Aplikasi yang dapat diakses dari berbagai perangkat klien. Konsumen bisa menggunakan *thin client*, atau *web browser* sebagai *interface*/antarmukanya.

Seperti *Amazon Web Services* (AWS) menyediakan *virtual server* dengan alamat IP yang unik dan blok penyimpanan sesuai permintaan. Pelanggan mendapatkan manfaat dari sebuah API dimana mereka dapat mengontrol server mereka. Karena pelanggan membayar sejumlah pelayanan yang mereka gunakan, seperti halnya membayar listrik atau air, dan layanan ini juga disebut utilitas komputasi.

**2. *Platform as a Service* (PaaS)**

Sistem PaaS mengijinkan pengguna menggunakan aplikasi dan bahasa pemrograman yang disediakan oleh sistem serta menyimpan data-data di dalam sistem *cloud computing*, tentunya dengan menggunakan API provider. *Google Apps* merupakan salah satu yang paling terkenal sebagai penyedia PaaS.

**3. *Software as a Service* (SaaS)**

SaaS merupakan perangkat lunak yang berbentuk layanan/*service*. Dalam hal ini *provider* memungkinkan pelanggan hanya untuk menggunakan aplikasi tersebut (aplikasi yang disewa). Perangkat lunak ini berinteraksi dengan *user* melalui *user interface*. Aplikasi ini dapat berupa *email* berbasis *web*, aplikasi seperti *Twitter* atau *Last.fm*.

**2.2.4. Perangkat Lunak *Cloud Computing***

**2.2.4.1. *Ubuntu Enterprise Cloud (UEC)***

Ubuntu merupakan salah satu distribusi sistem operasi berbasis linux yang dikembangkan dari linux Debian. Ubuntu server memiliki kelebihan yang tidak dimiliki oleh distribusi linux lainnya, yaitu meluncurnya versi terbaru ubuntu setiap enam bulan sekali yang disertai paket-paket aplikasi terkini juga penggunaan kernel linux terbaru. Sistem operasi Ubuntu Server dapat dipasang pada beberapa tipe arsitektur komputer diantaranya Intel X86, AMD64, ARM, SPARC, PowerPC, Itanium64 bahkan pada Playstation3.

*Ubuntu Server 10.04* alias *Lucid Lynx* yang merupakan distribusi ubuntu dengan fitur terintegrasi *Ubuntu Enterprise Cloud* (UEC). Ubuntu versi UEC ini meningkatkan stabilitas pada kemampuan *server cloud*, baik *cloud* dengan jaringan Amazon atau *cloud* yang dengan jaringan mandiri. *Eucalyptus* sebagai pendukung *cloud* ini juga ikut ditingkatkan. Jurnal Ibnu Yahya Sumantri,dkk (2010:2)

**2.2.4.2. *Proxmox***

*Proxmox* merupakan software *open source virtualization platform* untuk menjalankan *Virtual Appliances* dan *Virtual Machines*. *Proxmox* adalah distro khusus yang didekasikan secara khusus sebagai mesin host virtualisasi sistem dan memuat 2 teknologi virtualisasi, yaitu *KVM* dan *openVZ.* ([http://shop.tlab.co.id/89-proxmox diakses pada tanggal 05/12/12](http://shop.tlab.co.id/89-proxmox%20diakses%20pada%20tanggal%2005/12/12))

**2.2.4.3. *OpenStack***

*OpenStack* adalah teknologi *cloud computing* yang menyediakan sistem operasi *cloud* untuk *public* dan *private cloud* di bawah *Apache License*.Saat ini telah didukung oleh lebih dari 60 *company* yang berkontribusi untuk mengembangkan teknologi ini. (Ono W. Purbo, 2010:7)

**2.2.4.4. *OpenNebula***

OpenNebula VM Manager adalah sebuah komponen dasasr dari Reservoir. Ia adalah sebuah jawaban open-source untuk berbagai macam jenis virtual machine management yang banyak di gunakan secara proprietary, Interface nya pun dapat dengan mudah dipahami dengan cloud infrastructure tools and services. &ldquo;OpenNebula adalah sebuah open-source virtual infrastructure engine yang akan memberikan anda implementasi dan re-placement dari virtual machines pada physical resources,&rdquo; menurut project lead mereka. (Agrios Kanny)

**2.2.4.5. *Eucalyptus***

Eucalyptus adalah software yang tersedia di bawah GPL yang dapat menolong untuk membuat dan mengatur private Cloud maupun public Cloud. Eucalyptus menjadi sangat populer dan tampaknya telah menjadi salah satu kunci platform cloud open source. Karena Eucalyptus memungkinkan layanan tersedia melalui API yang compatible dengan layanan EC2 / S3, tool client di tulis untuk Amazon Web Services (AWS) dapat digunakan menggunakan Eucalyptus. (Ono W. Purbo, 2012:6)

**Gambar 2.1** Model Cloud berbasis Eucalyptus

**2.2.4.6. *KVM***

KVM adalah kependekan dari “Kernel Based Virtual Machine” adalah sebuah solusi untuk melakukan virtualisasi pada linux dengan hardware type x86 (64-bit). KVM memiliki keunggulan karena dapat diinstall pada sebuah linux OS minimum Server configuration ( No-GUI ) sehingga cenderung lebih ringan dibanding rekan-rekan aplikasi Virtual Machine yang lainnya . tetapi KVM cenderung memiliki kelemahan pada lambatnya proses I/O masing-masing Guest OS nya. ada lagi kelebihan KVM dibandig VM lainnya yaitu proses live migration. ( David E.Y Sarana, 2010:149)

**2.2.4.7. *QEMU***

*QEMU* adalah open source sebuah mesin emulator yang memungkinkan Anda untuk menjalankan hampir semua sistem operasi saat Anda sedang berjalan di dalam distribusi [*linux*](http://de-kill.blogspot.com/2009/05/konfigurasi-qemu-untuk-jalankan.html) lain. Dalam tutorial singkat ini saya akan menunjukkan bagaimana untuk menjalankan distribusi ISO yang tersedia, secara langsung dari dalam Debian atau Ubuntu. (David E.Y Sarana, 2010:150)

**2.2.4.8. *Xen***

*Xen hypervisor* dikembangkan dan dikelola oleh *Xen.org* masyarakat menyebutnya sebagai solusi gratis berlisensi di bawah *GNU General Public License*. Sebuah versi dari *Xen* yang digunakan oleh *Amazon EC2*. *xen* menyediakan dukungan untuk x86, x86-64, Itanium, Power PC, dan prosesor ARM, yang memungkinkan *Xen hypervisor* untuk berjalan pada berbagai perangkat komputasi. Saat Xen mendukung *Linux, NetBSD, FreeBSD, Solaris, Windows,* dan sistem operasi umum yang berjalan pada hypervisor.  
*Xen* dapat didownload tanpa biaya di <http://www.xen.org/produk/downloads.html>. *Citrix* juga menyediakan versi gratis dari *Xen*, yang disebutnya *XenServer*, yang mendukung versi 64-bit dari Intel dan prosesor AMD. *Citrix Essentials* untuk *XenServer* adalah ekstensi produk komersial *XenServer* (yang dalam domain publik, karena didasarkan pada teknologi *open source*). *Citrix Essentialsoffers* adalah kemampuan virtualisasi manajemen yang canggih ke server *public* untuk membantu pelanggan menciptakan kemudahan dalam hal pengelolaannya, dan infrastruktur maya yang lincah. Harga tergantung pada basis per-server dan termasuk dukungan selama satu tahun. *Symantec Corporation* juga menawarkan *Xenbased produk.* ( David E.Y Sarana, 2010:149-150 )

**2.2.5. Komponen dari *Cloud* Berbasis *Eucalyptus (UEC)***

**2.2.5.1. *Node Controller (NC)***

Sebuah node UEC adalah sebuah server dengan prosesor yang mempunyai kemampuan Virtualization Technology (VT) yang sudah di aktifkan yang mampu untuk menjalankan hypervisor seperti KVM. UEC akan secara automatis menginstalsi KVM jika pengguna memilih untuk menginstalasi node UEC. Virtual Machine (VM) yang dijalankan pada hypervisor dan di kontrol oleh UEC biasanya di sebut sebagai instance. Selain KVM, Eucalyptus juga mendukung hypervisor lain seperti Xen. Akan tetapi Canonical lebih memilih KVM sebagai hypervisor untuk UEC. Node Controller yang di jalankan di setiap node akan mengatur siklus kehidupan dari instance yang jalan di node. Di satu sisi, NC berinteraksi dengan sistem operasi dan hypervisor yang jalan di node. Di sisi lain, NC akan berinteraksi dengan CC. NC akan menanyakan sistem operasi yang jalan di node untuk mengetahui sumber daya fisik yang digunakan node, seperti, jumlah core, besar memory, ketersediaan disk dan juga mencek status dari VM instance yang jalan di node dan memberikan informasi tersebut ke CC.

Fungsi:

1. Mengumpulkan data yang terkait dengan ketersediaan dan penggunaan sumber daya di node dan melaporkan ke CC.

2. Manajemen siklus kehidupan dari instance. (Ono W. Purbo, 2012:7)

**2.2.5.2. *Cluster Controller (CC)***

CC memanage satu atau lebih Noce Controller (NC) dan menjalankan / memanaje instance pada NC. CC juga memanage networking untuk instance yang jalan di Node sesuai dengan permintaan mode jaringan dari Eucalyptus. CC berkomunikasi dengan CLC di satu sisi dan banyak NC di sisi lain.

Fungsi:

1. Menerima permintaan dari CLC untuk menjalankan instance.

2. Memutuskan NC yang mana yang digunakan untuk menjalankan instance tersebut.

3. Mengatur virtual network untuk instance.

4. Mengumpulkan informasi tentang NC yang terdaftar dan melaporkannya ke CLC. (Ono W. Purbo, 2012:8)

**2.2.5.3. *Walrus Storage Controller (WS3)***

WS3 memberikan layanan penyimpanan yang sederhana tapi 'ngotot' / presistent menggunakan API REST dan SOAP yang kompatibel dengan API S3.

Fungsi:

1. Menyimpan machine image (sistem operasi untuk instance).

2. Menyimpan snapshot

3. Menyimpan dan memberikan layanan file menggunakan API S3.

WS3 dapat dilihat sebagai sebuah sistem penyimpanan file yang sederhana. (Ono W. Purbo, 2012:9)

**2.2.5.4. *Storage Controller (SC)***

SC menyediakan tempat penyimpanan (storage) block dimana instance akan melihatnya sebagai harddisk. Layanan ini mirip dengan layanan Elastic Block Storage (EBS) dari AWS.

Fungsi:

1. Pembuatan dari device EBS

2. Memberikan layanan block storage melalui protokol AoE atau iSCSI ke instance

3. Memungkinkan pembuatan snapshot untuk volume. (Ono W. Purbo, 2012:9)

**2.2.5.5. *Cloud Controller (CLC)***

Cloud Controller (CLC) adalah front end dari seluruh infrastruktur cloud. CLC memberikan antar muka layanan web yang compliant dengan EC2 / S3 ke client di satu sisi. Di sisi lain, CLC berinteraksi dengan seluruh komponen infrastruktur Eucalyptus. CLC memberikan antar muka web ke user untuk melakukan manajemen beberapa aspek dari infratruktur UEC.

Fungsi:

1. Memonitor ketersediaan sumber dari di berbagai komponen infrastruktur cloud, termasuk hypervisor pada node yang digunakan untuk melakukan manajemen instance dan cluster controller untuk melakukukan manajemen pada node hypervisor.

2. Arbitrasi Sumber Daya - menentukan cluster mana yang akan digunakan untuk bertanggung jawab sebuah instance.

3. Memonitor instance yang sedang berjalan.

Pendek kata, CLC mempunyai pengetahuan yang lengkap tentang ketersediaan dan penggunaansumber daya di cloud maupun status cloud. Eucalyptus Cloud. (Ono W. Purbo, 2012:9)

**BAB III**

**TINJAUAN OBJEK**

**3.1 Sejarah UPT-SIM Universitas Bina Darma**

UPT-SIM merupakan unit pelaksana teknis yang memberikan dukungan kepada Universitas Bina Darma (UBD) dalam hal pengembangan dan penyediaan layanan ICT. Layanan yang diberikan pada saat ini adalah layanan pengolahan data dan penyediaan akses intra dan internet untuk seluruh civitas akademika UBD. Pada saat ini UPT-SIM memiliki divisi antara lain : divisi pengembangan sistem, divisi pengolahan data, divisi *maintenance* dan divisi *network administrator*. Setiap divisi memiliki tugas masing-masing seperti, divisi pengembangan sistem memiliki tugas untuk mengembangkan atau membangun sistem untuk mendukung proses akademik dan non akademik di UBD. Divisi pengolahan data bertugas untuk mengolah data akademik (pendukung proses belajar mengajar) dan proses pelaporan EPSBED / PDPT. Divisi maintenance bertugas untuk memelihara perangkat keras komputer dan jaringan komputer dilingkungan UBD. Divisi *network administrator* bertugas untuk memberikan layanan koneksi jaringan, *routing*, mengelolah *web* domain dan pemeliharaan *server*.

Layanan yang telah diberikan UPT- SIM untuk mendukung proses akademik dan non akademik dilingkungan UBD antara lain : UPT-SIM telah mendukung proses komputerisasi dengan membangun sistem akademik yang meliputi : proses penjadwalan, entri data registrasi mahasiswa baru dan lama, entri data nilai, KHS *Online* dan KRS intranet. Selain dari itu, UPT-SIM telah membagun sistem perhitungan KIDO/KIKAR, HRIS, Laman binadarma.ac.id , layanan *email*, *e*-*Learning* dan *hosting* untuk beberapa unit kerja seperti perpustakaan (otomasi.binadarma.ac.id dan digilib.binadarma.ac.id), ELC (elc.binadarma.ac.id), BDCTC (bdctc.binadarma.ac.id), laman *web* fakultas dan beberapa laman *web* program studi. Lebih lanjut UPT-SIM juga telah memberikan layanan akses data / internet bagi staf/dosen dan mahasiswa melalui kabel dan nirkabel (*wireless* *WIFI*) yang dapat diakses diseluruh lingkungan UBD.

**3.2 Visi dan Misi UPT-SIM Universitas Bina Darma**

**3.2.1 Visi UPT-SIM Universitas Bina Darma**

Menjadi salah satu unit kerja di Universitas Bina Darma yang mendukung implementasi pelayanan teknologi informasi secara optimal untuk mewujudkan universitas bestandar internasional berbasis teknologi informasi pada tahun 2025.

**3.2.2 Misi UPT-SIM Universitas Bina Darma**

Memberikan pelayanan informasi terbaik dan terbaru bagi semua unit di Universitas Bina Darma, dengan menerapkan pengembangan teknologi informasi yang terbaru.

**3.2.3 Tujuan UPT-SIM Universitas Bina Darma**

UPT-SIM terus meningkatkan kinerjanya untuk mencapai tujuan berikut :

1. Menjadi pusat pengelolahan data yang handal
2. Menjadi pusat penggunaan teknologi informasi
3. Menjadi pusat pengembangan teknologi informasi

**3.4 *Job Description* UPT-SIM Universitas Bina Darma**

Dari Gambar 3.1 dapat dipaparkan mengenai *job description* dari masing-masing sub-unit yang ada di UPT-SIM Universitas Bina Darma, seperti yang tercantum dibawah ini :

1. Kepala UPT-SIM Universitas Bina Darma

Kepala UPT-SIM Bertanggung jawab atas seluruh kegiatan *Computer Units Technical Support (CUTS)*. Adapun yang menjadi tanggung jawab kepala UPT-SIM adalah sebagai berikut :

1. Bertanggung jawab kepada pimpinan perguruan tinggi terhadap pelaksanaan tugasnya serta lingkungan yang ada di UPT-SIM.
2. Bertanggung jawab terhadap kelancaran pelaksanaan sistem informasi.
3. Sub-unit Pengembangan Sistem

Sub-unit pengembangan sistem adalah sub-unit yang membantu kepala UPT-SIM dalam mengembangkan dan meningkatkan pelayanan sistem informasi. Adapun yang menjadi tanggung jawab sub-unit pengembangan sistem adalah bertanggung jawab langsung kepada kepala UPT-SIM atas semua pekerjaan mengenai kinerja sistem informasi antara lain :

1. Sistem informasi manajemen
2. Sistem informasi akademik
3. Sistem informasi akutansi dan keuangan
4. Integrasi sistem informasi secara keseluruhan
5. Konektifitas perangkat lunak sistem terhadap semua unit pelaksana dan unit pelayanan.
6. Sub-unit Pengelolahan Data

Sub-unit pengelolahan data adalah sub-unit yang bertanggung jawab dalam hal mengelolah data dan merawat basis data. Adapun yang menjadi tanggung jawab sub-unit pengelolahan data adalah bertanggung jawab kepada kepala UPT-SIM didalam pengadaan, penyeleksian dan penyediaan data untuk semua bagian dan unit pelayanan yang ada, antara lain :

1. Terhadap administrasi akademik dengan memberikan dukungan data kepada Pusat Pelayanan Mahasiswa (PPM), dosen, dan mahasiswa dalam proses pendidikan.
2. Terhadap administrasi akademik dengan memberikan dukungan data kepada bagian keuangan dalam proses akademik.
3. Mengamankan dan memvalidasi data, sehingga dapat menghasilkan informasi yang sesuai dan akurat.
4. Sub-unit Perawatan dan Perbaikan

Secara umum sub-unit perawatan dan perbaikan adalah sub-unit yang melakukan perawatan dan perbaikan terhadap fasilitas komputasi dan infrastruktur teknologi informasi. Adapun yang menjadi tanggung jawab sub-unit perawatan dan perbaikan adalah bertanggung jawab kepada kepala UPT-SIM didalam perawatan dan perbaikan perangkat komputasi, LAN, dan sistem operasi pada semua komputer yang terhubung ke *server*.

1. Sub-unit *Network Operation Centre*

Secara umum sub-unit *network operation centre* adalah sub-unit yang mengelolah, menjaga dan merawat situs web. Adapun yang menjadi tanggung jawab dari sub-unit *network operation centre* adalah bertanggung jawab kepada kepala UPT-SIM terhadap :

1. Kelancaran koneksi internet
2. Mengelolah akses koneksi intranet (*user administration*)
3. Menjaga trafik internet
4. Pemuktahiran halaman situs
5. Perawatan *mail server*
6. Keamanan situs.

**3.5 Sistem yang Sedang Berjalan di UPT-SIM**

Universitas Bina Darma adalah salah satu universitas yang menggunakan teknologi informasi dan komunikasi dalam sarana mengembangkan setiap kegiatan yang berlangsung didalam universitas Bina Darma. Banyak aplikasi yang berjalan di universitas Bina Darma, salah satunya dalam proses kegiatan belajar mengajar antar mahasiswa dan dosen, universitas Bina Darma sudah menerapkan *e-learning* dan *zimbra-mail* dalam menunjang kegiatan tersebut. Akan tetapi sistem-sistem yang telah berjalan tersebut belum terintegrasi, setiap sistem yang berjalan di UPT-SIM Universitas Bina Darma memiliki server dan database yang berbeda dan juga hal ini tentunya membutuhkan *login* di masing-masing sistem yang berjalan tersebut. Untuk melakukan pengamanan UPT-SIM Universitas Bina Darma telah meletakkan Radius server, yang mana Radius server ini berfungsi sebagai penjembatan untuk memaksa user kehalaman login hotspot Universitas Bina Darma terlebih dahulu yang kemudian akan dilakukan autentikasi dan autorisasi terhadap pengguna. Jika berhasil maka user akan dapat mengakses sistem lainnya dan tentunya harus login terlebih dahulu dimasing-masing sistem tersebut.

(Sumber : UPT-SIM Universitas Bina Darma)

Gambar 3.2 Skema jaringan UPT-SIM Universitas Bina Darma

Mulai

Mulai

No No

Yes Yes

Cek autentikasi di database

*Elearning*

*Elearning*

*Mail*-zimbra

*Login*

*Login*

*Login*

*Login*

Cek autentikasi di database

Menggunakan Layanan

Men

Menggunakan Layanan

Menggunakan Layanan

Keluar

Keluar

Berhenti

Berhenti

Gambar 3.3 Diagram Alir sistem saat ini di UPT-SIM Universitas Bina Darma

**3.4.1 Daftar Server yang Berjalan di UPT-SIM Universitas Bina Darma**

Tabel 3.1 Daftar *Server* di Universitas Bina Darma

|  |  |  |
| --- | --- | --- |
| NO. | Nama Server | ISI |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15. | Server Proxy Gateway Internet  Server Radius  Server Proxy Radius  Server Intranet1  Server DNS 4.3  Server Blog  Server Website  Server Mail  Server Repository &  FTP  Server VPN  Server E-Learning  Server E-Learning Director  Server Load Balancing E-Learning  Server Load Balancing E-Learning  Server Fakultas dan Aplikasi | Ns1.Binadarma.ac.id 10.237.2.2  Radius Server Authentification, Chilispot  Database User  PMB, Intranet1, Entry KRS   1. <Intranet1.binadarma.ac.id/entrykrs> 2. Intranet1.binadarma.ac.id/spmb 3. Intranet.binadarma.ac.id   Intranet, DNS 10.237.4.3  Website Blog (blog.binadarma.ac.id)  Website Utama (www.binadarma.ac.id)  Mail Server (mail.binadarma.ac.id)  Repository (repo.binadarma.ac.id) & FTP (ftp://ftp.binadarma.ac.id)  VPN PMB  Website Database (Elearning.binadarma.ac.id)  Pembagi Load Balancing  Content Elearning  Content Elearning   1. elc.binadarma.ac.id 2. ikati.binadarma.ac.id 3. teknikelektro.binadarma.ac.id 4. ekonomi.binadarma.ac.id 5. hris.binadarma.ac.id |

|  |  |  |
| --- | --- | --- |
| NO. | Nama Server | ISI |
|  |  | 1. kd-bidar.binadarma.ac.id 2. ilkom.binadarma.ac.id 3. pmb.binadarma.ac.id 4. sap.binadarma.ac.id 5. reg.binadarma.ac.id 6. semnastik.binadarma.ac.id 7. if.binadarma.ac.id 8. bdctc.binadarma.ac.id 9. pbi.binadarma.ac.id 10. io.binadarma.ac.id 11. si.binadarma.ac.id 12. tk.binadarma.ac.id 13. md.binadarma.ac.id 14. ka.binadarma.ac.id 15. fekon.binadarma.ac.id 16. mn.binadarma.ac.id 17. ak.binadarma.ac.id 18. mp.binadarma.ac.id 19. ab.binadarma.ac.id 20. sastra.binadarma.ac.id 21. bi.binadarma.ac.id 22. teknik.binadarma.ac.id 23. ts.binadarma.ac.id 24. te.binadarma.ac.id 25. in.binadarma.ac.id 26. fkip.binadarma.ac.id 27. po.binadarma.ac.id 28. psikologi.binadarma.ac.id 29. ps.binadarma.ac.id 30. mti.binadarma.ac.id 31. mm.binadarma.ac.id |

Tabel 3.1 Lanjutan

Tabel 3.1 Lanjutan

|  |  |  |
| --- | --- | --- |
| NO. | Nama Server | ISI |
| 16.  17.  18. | Server Database  Server Otomasi  Server Ganjil / Genap | 1. fikom.binadarma.ac.id 2. ik.binadarma.ac.id 3. lppm.binadarma.ac.id 4. pasca.binadarma.ac.id 5. bdec.binadarma.ac.id 6. inventory.binadarma.ac.id 7. simperang.binadarma.ac.id 8. iso.binadarma.ac.id 9. foresec.binadarma.ac.id 10. cisco.binadarma.ac.id 11. ejournal.binadarma.ac.id 12. scholar.binadarma.ac.id 13. openjournal.binadarma.ac.id 14. digilib.binadarma.ac.id   Database  Server Otomasi Perpustakaan  Server Perol dan Ganjil / Genap |

(Sumber: UPT-SIM Universitas Bina Darma)

**3.4.2 Spesifikasi Alat yang Digunakan di UPT-SIM Universitas Bina Darma**

Tabel 3.2 Spesifikasi Alat yang Digunakan di Universitas Bina Darma

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Nama Barang / Peralatan | Seri | Pabrikasi |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17.  18  1.  2.  3.  4.  5.  1. | **KOMPUTER SERVER**  Server Proxy Gateway  Internet  Server Radius  Server Proxy Radius  Server Intranet1  Server DNS 4.3  Server Blog  Server Website  Server Mail  Server Repository & FTP  Server VPN  Server E-Learning  Server E-Learning Director  Server Load Balancing E-Learning  Server Load Balancing E-Learning  Server Fakultas  Server Database  Server Otomasi  Server Ganjil / Genap  **U.P.S**  UPS Ke Seluruh Server  UPS Ke Server #1  UPS Ke Server #2  UPS Ke Router & Modem  UPS ke Switch Catalyst  UPS Ke Server #7  **MODEM & ROUTER**  Router Akses Lokal & Ke | System X3400 M3 Xeon 2,4 Ghz  System X3100 Xeon 1,6 Ghz  System X3100 Xeon 1,6 Ghz  Assembling Computer P.III 800 Mhz  Assembling Computer P.III 800 Mhz  System X3100  System X3200 M2 Xeon Xeon 2,5 Ghz  System X3400 M3 Xeon 2,4 Ghz  Sytem X3100 M3 Xeon 2,6 Ghz  Assembling Computer P.IV 2,6 Ghz  System X3250 M3 16Gb Xeon 3,0  Sytem X3100 M3 Xeon 2,6 Ghz  System X3250 M3 Xeon 3,0  System X3250 M3 Xeon 3,0  System X3250 M3 4 Gb Xeon 3,0  System X3250 M3 16Gb Xeon 3,0  Assembling Computer AMD AThlon 64FX  System X3250 M3 Xeon 3,0  ICA SIN 3100 C  ICA 608 B  ICA 608 B  ICA 608 B  ICA 608 B  ICA CT1682 B  Router Cisco 2800 | IBM  IBM  IBM  Gigabyte  Gigabyte  IBM  IBM  IBM  IBM  Gigabyte  IBM  IBM  IBM  IBM  IBM  IBM  Gigabyte  IBM  ICA  ICA  ICA  ICA  ICA  ICA  Cisco |

Tabel 3.2 Lanjutan

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Nama Barang / Peralatan | Seri | Pabrikasi |
| 2.  3.  1.  2.  3.  4.  5.  6.  7.  8.  9.  1.  2.  3.  1.  2. | Inherent  Mikrobit Ainos Master Vlan  Mikobit Aneto VPN  **SWITCH, KVM & PANEL**  Patch Panel  Patch Panel  Switch  Switch  Switch  Switch  KVM ATEN  KVM Monitoring 1  KVM Monitoring 2  **MONITOR**  LCD untuk KVM #ATEN  LCD untuk KVM #1  LCD untuk KVM #2  **AIR CONDITIONER**  AC PANASONIC  AC PANASONIC | Ainos 2071  Aneto  AMP Netcome CT  AMP Netcome CT  Switch Cisco Catalyst 2950  Switch Cisco Catalyst 3650  Switch Cisco Catalyst 2960  Switch Cisco Catalyst 2960  KVM CS1758 8 Port  KVM 0420 4 Port USB  KVM 0420 4 Port PS2  SVGA 14 Inch  SVGA 14 Inch  SVGA 15 Inch  TAC-09CS 220-240 Volt ; 950 Watt  TAC-09CS 220-240 Volt ; 950 Watt | Mikrotik  Mikrotik  AMP  AMP  Cisco  Cisco  Cisco  Cisco  ATEN  Level One  Level One  Ion  Ion  Ion  Panasonic  Panasonic |

(Sumber: UPT-SIM Universitas Bina Darma)

Dari banyaknya sistem yang berjalan di universitas Bina Darma tersebut, sebagian besar belum dilakukan integrasi. Seperti misalnya pada sistem *e-learning* dan *zimbra-mail*. Kedua sistem ini membutuhkan database masing-masing dan juga membutuhkan *login* (*username* dan *password*). Sebagai sistem autentikasinya universitas Bina Darma menggunakan Radius Server, yang juga membutuhkan database dan login sendiri.

**BAB IV**

**METODOLOGI PENELITIAN**

**4.1. Melakukan Kajian Terhadap Sistem yang Sedang Berjalan**

Melakukan kajian secara induktif yang berkaitan erat dengan objek penelitian. Mengidentifikasi komponen-komponen apa saja yang digunakan dalam penelitian. Pada langkah pertama ini peneliti mengkaji sistem yang sedang berjalan di Universitas Binadarma. Penjelasan terhadap sistem yang sedang berjalan di Universitas Binadarma telah di jelaskan pada Tabel 3.1 Daftar *Server* di Universitas Bina Darma

**4.2. Identifikasi Permasalahan Terhadap Sistem yang Sedang Berjalan**

Pada Universitas Bina Darma juga telah memiliki banyak sekali sistem informasi yang digunakan untuk mendukung aktifitasnya. Salah satunya adalah sistem informasi Pasca Sarjana Universitas Bina Darma. Server konvensional akan di batasi oleh jumlah core processor, harddisk dan memory. Dengan keterbatasan fisik yang ada maka kita tidak mungkin membebani sebuah server konvensional dengan beban maksimal. Jika resource / sumber daya habis, maka biasanya kita harus menginstall ulang seluruh aplikasi dan data di server yang kapasitasnya lebih besar dan memigrasi semua aplikasi yang ada ke server yang baru. Ini akan membutuhkan waktu 1-2 hari untuk menyiapkan sebuah server baru, itupun kalau tidak ada masalah. Data pada sistem tersebut semakin lama akan semakin bertambah dan itu membuat diperlukannya tempat penyimpanan atau *storage* yang besar. Dengan semakin bertambahnya data-data pada sistem tersebut dapat pula mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya semakin bertambah.

Oleh karena itu, salah satu solusi permasalahan tersebut yaitu menyediakan tempat penyimpanan atau storage yang besar, yang tidak mengakibatkan pekerjaan dan biaya pemeliharaan serta perawatan perangkat kerasnya menjadi bertambah.*cloud computing* ditunjuk sebagai teknologi yang dapat mengatasi masalah tersebut. Teknologi ini menggabungkan prinsip dasar ekonomi dan peletakan sumber daya komputasi. Sesuai dengan karakteristiknya yaitu virtualisasi sumber daya komputasi dan penyewaan berbasis pengguna. Dalam hal ini penggunaannya pada sistem informasi Pasca Sarjana di Universitas Bina Darma dapat menggunakan *private cloud computing* menggunakan *eucalyptus-ubuntu enterprise cloud.* Yang mana merupakan pemodelan *cloud computing* yang memberikan lingkup yang lebih kecil untuk dapat memberikan layanan kepada pengguna tertentu

**4.3. Studi Literatur Sisitem di Universitas Binadarma**

Spesifikasi alat yang sedang berjalan pada Universitas Binadarma, telah di jelaskan pada Tabel 3.2 Spesifikasi Alat yang Digunakan di Universitas Bina Darma

**4.4. Rencana Kegiatan Penerapan *Private Cloud Computing***

Kebutuhan perangkat keras untuk pembangunan *server cloud* sangat tinggi, dari segi *proccesor*, *memory* dan *storage* mempunyai kebutuhan yang sangat tinggi. Untuk membangun sebuah *server cloud* dibutuhkan dau buah *server*, sesuai dengan standart pembangunan *server cloud* yang menggunakan aplikasi Eucalyptus. *Server* satu digunakan untuk installasi *Cloud Controller*, *Cluster Controller*, *Walrus* *Controller* dan *Storage Controller* mempunyai kebutuhan yang tidak terlalu tinggi. *Server* satu juga digunakan sebagai *web interface* penghubung antara *client* dan *server cloud*.

Sedangkan untuk *server* dua yang digunakan untuk *Node Controller* mempunyai kebutuhan yang sangat tinggi karena menentukan hasil vm yang biasdibangun untuk *web server*. Dari *processor*, *memory*, dan *disk storage* yangdigunakan mempunyai spesifikasi yang tinggi serta CPU yang digunakan harus sudahmendukung untuk *Virtual Technology* karena pada *server* dua ini digunakan untuktempat VM atau *instance* yang akan dijalankan. Semakin baik *processor*, *memory* dan *disk storage* maka akan semakin banyak VM atau *instance* yang bisa dijalankandi *Node Controller*. Sedangkan *PC* Admin berfungsi untuk membuat, *bundling* dan meregistrasikan *image* yang baru..

**4.4.1. Spesifikasi Sistem**

**4.4.1.1 Spesifikasi Server I**

Pada pengujian yang dilakukan, spesifikasi sistem yang dilakukan seperti pada table 4.1.

Tabel 4.1 Spesifikasi Server I

|  |  |
| --- | --- |
| Keterangan | Spesifikasi |
| Perangkat Lunak | * Ubuntu 10.04 LTS Server Edition * Eucalyptus |
| Perangkat Keras : |  |
| CPU | * *Intel Core 2 Duo Processor T 6600* |
| Memory | * 2 GB |
| *Hard Disk* | * 80 GB |

**4.4.1.2 Spesifikasi Server II**

Pada pengujian yang dilakukan, spesifikasi sistem yang dilakukan seperti pada table 4.2.

Tabel 4.2 Spesifikasi Server II

|  |  |
| --- | --- |
| Keterangan | Spesifikasi |
| Perangkat Lunak | * Ubuntu 10.04 LTS Server Edition * Eucalyptus |
| Perangkat Keras : |  |
| CPU | * *Intel Core I5 3470 3.2 GHZ LGA 1155 BOX* |
| Memory | * 4 GB |
| *Hard Disk* | * 250 GB |

**4.4.1.3 Spesifikasi Client**

Pada pengujian yang dilakukan, spesifikasi sistem yang dilakukan seperti pada table 4.3.

Tabel 4.3 Spesifikasi Client

|  |  |
| --- | --- |
| Keterangan | Spesifikasi |
| Perangkat Lunak | * Ubuntu 10.10 Deskstop Edition |
| Perangkat Keras : |  |
| CPU | * *Intel Core 2 Duo Processor T 6600* |
| Memory | * 2 GB |
| *Hard Disk* | * 250 GB |

**4.4.2. Rancangan Sistem**

Gambar 4.1 Rancangan Sistem

Gambar 4.1 menunjukan desain fisik sistem *cloud* yang akan dibangun. *Design* adalah tahap dimana terdapat penggambaran tentang pengguna dan *design* topologiyang diimplimentasikan pada *cloud* jenis *private* dengan layanan *IaaS* yangmenggunakan aplikasi Eucalyptus. Terdapat dua *user* yang terdapat pada sistem ini,yaitu admin dan *client*, *user* admin berfungsi untuk membuat *user* atau *client* baru,memonitoring *server cloud*, manajemen *remote access* seluruh *client*, dan memonitor *instance* seluruh *client*. *User* kedua adalah *client* sebagai pengguna *service cloud* mempunyai fungsi yaitu membuat *instance* dengan *image* yang telah tersedia,memonitor dan mematikan *instance client* itu sendiri, serta manajemen *remote access* agar *client* bisa berkomunikasi dengan *instance* yang *client* itu dibuat. Pada *fase design* juga membahas tentang penggambaran tentang desain detil secara logis dariperancangan infrastruktur yang sesuai dengan mekanisme sistem *IaaS*.

**4.5. Melakukan Eksperimen**

Pada tahapan ini akan dijelaskan tentang mengimplementasikan eksperimen dengan harapan dapat menyelesaikan masalah.

## 4.5.1. Proses Instalasi

Pada bagian ini akan dijelaskan tentang proses pengintegrasian aplikasi yang akan di gunakan di Universitas Bina Darma. Pengintegrasian aplikasi ini memerlukan tahap-tahap yang harus dilalui terlebih dahulu. Tahap-tahap ini dijelaskan pada sub bab di bawah ini.

**4.5.1.1. Instalasi UEC Cloud Controller**

Proses instalasi akan membutuhkan waktu sekitar 30-50 menit-an. Detail langkah adalah sebagai berikut:

• Masukan CDROM Ubuntu Server-server

• Boot Komputer

• CDROM Booting. Pilih bahasa.

• Pilih “Install Ubuntu Enterprise Cloud”

• Pilih bahasa.

• Pilih negara “other”

• Pilih negara “Asia”

• Pilih negara “Indonesia”

• Detect Keyboard Layout -pilih “No”

• Origin of Keyboard “USA”

• keyboard layout “USA”

• Pilih “configure network manually”

• IP address – masukan IP cloud controller 192.168.43.1

• Netmask – masukan 255.255.255.0

• Gateway – masukan 192.168.43.1

• hostname – misalnya “cloudcontroller”

• domain name – bisa kosong

• Cloud Controller Address – bisa kosong, kalau kita satu-satunya cloud controller.

• Select cloud installation mode pilih Cloud Controller, Walrus Storage Service, Cluster Controller dan Storage Controller.

Gambar 4.2 Proses Instalasi UEC Cloud Controller

• Configure the clock – pilih “Jakarta”

• Partision Disk – kalau tidak mau pusing pakai seluruh harddisk. Setelah selesai tekan “Finish Partitioning and Write Changes to Disk”

• Setup Users and Password

• Full name for the new user.

• Username for your account.

• Choose a password for the new user.

• Re-enter password to verify.

• Encrypt your home directory – pilih No kalau masih belajar.

• Configure the package manager – tekan continue.

• Configure tasksel – No Automatic update kalau sedang belajar.

• System mail name – misalnya cloud.perusahaan.com

• Eucalyptus Cluster name – misalnya cluster1

• Provide list of IP public – misalnya 192.168.33.151-192.168.33.200 untuk private cloud.

• Install GRUB boot loader – Yes.

• Finish Installation - pilih “Continue”.

**4.5.1.2. Instalasi UEC Node Controller**

Proses instalasi untuk membuat Node Controller lebih cepat daripada Cloud Controller. Akan membutuhkan waktu sekitar 15 menit-an. Detail langkah adalah sebagai berikut:

• Masukan CDROM Ubuntu Server-server

• Boot Komputer

• CDROM Booting. Pilih bahasa.

• Pilih “Install Ubuntu Enterprise Cloud”

• Pilih bahasa.

• Pilih negara “other”

• Pilih negara “Asia”

• Pilih negara “Indonesia”

• Detect Keyboard Layout -pilih “No”

• Origin of Keyboard “USA”

• keyboard layout “USA”

• Pilih “configure network automatically”

• hostname – misalnya “node1”

• domain name – bisa kosong

• Cloud Controller Address – 192.168.43.1 (sesuai dengan Cloud Controller kita).

• Select cloud installation mode pilih Node Controller.

Gambar 4.3 Proses Instalasi UEC Node Controller

• Configure the clock – pilih “Jakarta”

• Partision Disk – kalau tidak mau pusing pakai seluruh harddisk. Setelah selesai tekan “Finish Partitioning and Write Changes to Disk”

• Setup Users and Password

• Full name for the new user.

• Username for your account.

• Choose a password for the new user.

• Re-enter password to verify.

• Encrypt your home directory – pilih No kalau masih belajar.

• Configure the package manager – tekan continue.

• Configure tasksel – No Automatic update kalau sedang belajar.

• Install GRUB boot loader – Yes.

• Finish Installation - pilih “Continue”.

**BAB V**

**HASIL DAN PEMBAHASAN**

## 5.1. Hasil Eksperimen

Pada tahapan ini akan dijelaskan tentang hasil yang di dapat pada uji coba yang di akukan.

## 5.1.1 Hasil Instalasi Server

Pada bagian ini akan dijelaskan hasil dari proses instalasi server, yaitu hasil dari server 1 sebagai cloud controller dan server 2 sebagai node controller. Karena antara server 1 dan server 2 harus terhubung satu sama lain, adapun langkah-langkahnya sebagai berikut:

1. Masukan username dan password pada masing-masing server

Gambar 5.1 tampilan login cloud controller

1. Periksa alamat ip masing – masing server , jalankan perintah ifconfig

Gambar 5.2 ip server cloud controller

Gambar 5.3 ip server node controller

1. Pada server node, gunakan perintah ping untuk memastikan server node terhubung ke server cloud.

Gambar 5.4 server node terhubung pada server cloud

### 5.1.2. Hasil dan Konfigurasi pada PC Admin

Pada bagian ini akan dijelaskan hasil dan proses konfigurasi pada PC admin agar sistem dapat berjalan.

**5.1.2.1. Cek Ketersediaan Zone**

Untuk melihat ketersediaan zone yang ada pada cloud dapat menggunakan perintah

. ~/.euca/eucarc

source ~/.euca/eucarc

eucadescribeavailabilityzones

verbose



Gambar 5.5 Ketersediaan Zone

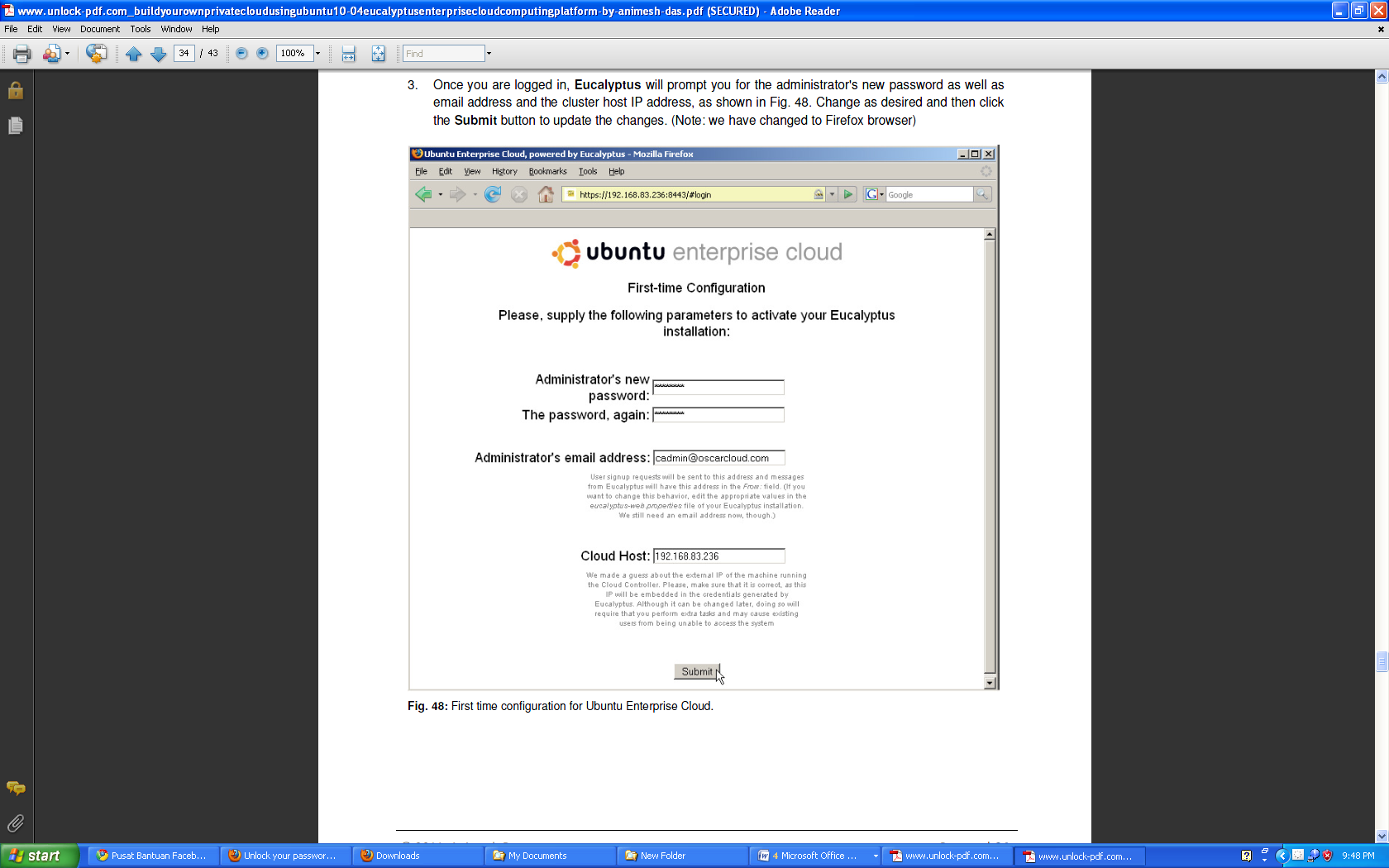
**5.1.2.2. Mengakses Web Antarmuka *Server Cloud***

Sebelum dapat mengakses dan menggunakan fasilitas yang ada pada *private cloud* ini *user/client* diwajibkan untuk melakukan registrasi, adapun langkah – langkahnya sebagai berikut.

1. Mengakses alamat IP dari *front end https://192.168.33.252:8443/* melalui *web browser*.
2. Akan muncul halaman login, buat username dan password yang akan digunakan.

Gambar 5.6 halaman login

1. Setelah Anda login, Eucalyptus akan meminta untuk membuat password baru, mengisi alamat email dan *cloud host.*



Gambar 5.7 Konfigurasi awal pada *cloud*

1. Setelah tekan submit, akan masuk ke halaman UEC

Gambar 5.8 Halaman UEC

1. Pada tab storage, install Ubuntu 9.10 - Karmic Koala (i386)

Gambar 5.9 install Ubuntu 9.10 - Karmic Koala (i386)

1. Pada ekstra tab di Enterprise Cloud konsol, menunjukkan Images Eucalyptus-bersertifikat yang siap untuk menginstal.

Gambar 5.10 Tab Extras

1. Sekarang pada tab images, telah terisi image yang kita butuhkan.

Gambar 5.11 Tab Images

**5.1.2.3. Manajemen User**

Admin dapat menentukan *account* mana saja yang diterima atau tidak, karena registrasi yang dilakukan *client* harus menunggu persetujuan dari admin.

******

Gambar 5.12 *User Management*

Untuk menerima akun yang telah melakukan registrasi yaitu dengan klik *Approve* pada kolom *Actions*. Akan tetapi cara ini membutuhkan konfirmasi dari pemilik akun yang bersangkutan, yaitu melalui *email* yang telah dituliskan pada *form* saat melakukan registrasi. Konfirmasi dapat dilakukan tanpa perlu membuka email, yaitu dengan cara klik *Edit* pada kolom *Actions*. Maka akan muncul tampilan seperti pada gambar, kemudian beri tanda cek pada “*Skip email confirmation*” kemudian klik tombol “*Update Record*”.



Gambar 5.13 *Approve Acount*

Selain user dapat melakukan registrasi secara langsung, untuk menambahkan akun dapat dilakukan oleh admin. Dengan klik “*Add user*” pada tab “*Users*” kemudian akan muncul halaman seperti gambar. Setelah lengkap diisi kemudian klik “*Add user*”.

Gambar 5.14 Add User dari Admin

Admin juga dapat dapat menonaktifkan akun yang sudah ada, yaitu dengan klik *Delete* pada kolom *Actions*.

**5.1.2.4. Membuat *Keypairs***

Pembuatan keypair ini menggunakan add-ons mozila firefox yaitu hybridfox. *Hybridfox* akan menampilkan konfigurasi apa saja yang dapat dilakukan oleh *user*, sesuai dengan *credential* dari masing-masing *account*. Ada beberapa cara untuk menginstansisasi sebuah *image* di UEC :

a. Menggunakan *Command Line*

b. Menggunakan *management tools* seperti *Landscape*

c. Menggunakan *hybridfox* yang merupakan *addons* dari *Mozila Firefox*.

Pada pembuatan sistem *private cloud* ini, manajemen terhadap *cloud* menggunakan *hybridfox* dengan dibantu konfigurasi dari *command line*. Sebelum menggunakan *hybridfox* pastikan bahwa *hybridfox* telah terinstal pada *addons mozila firefox*. Untuk mendapatkan *hybridfox*, dapat di unduh dari *link http://code.google.com/p/hybridfox/downloads/list*.

Langkah pertama untuk memulai *hybridfox* yaitu dengan menjalankan *mozila firefox* kemudian masuk pada menu *tools*. Kemudian *instances* yang akan dibuat atau yang telah ada akan secara automatis termonitor sesuai dengan *credential*s yang telah diberikan kepada masing-masing *user*. Langkah yang perlu dilakukan adalah :

1. Membuat *Regions*

Klik pada tombol “*Regions*” kemudian isi “*Region Name*” dengan apa saja. Dan untuk “*Endpoint URL*” isi dengan *http://<cloud-ip-address>:8773/services/Eucalyptus*. Kemudian klik tombol “*Add*” dan “*Close*”.

1. Membuat *Credentials*

Buka tab baru dan akses *https://<cloud-ip-address>:8443*. Buka tab “*Credentials*” dan klik “*Show keys*” pada “*Query interface credentials*”.



Gambar 5.15 *Download Credentials*

Kemudian klik pada tombol “*Credentials*”, isi “*Account Name*” dengan apa saja, “*AWS Access Key*” dengan “*Query ID*” dan “*AWS Secret Key*” dengan “*Secret Key*”. Klik “*Add*” kemudian “*Close*”.

1. Setelah langkah diatas maka *hybridfox* akan mendeteksi apa saja yang telah dikonfigurasi kepada *cloud* yang telah didaftarkan. Mulai dari ketersediaan *Image*, status *Instances*, *Elastic IPs*, *Snapshot*, *Security Groups* dan lainnya.

Pembuatan keypair dapat dilakukan dengan klik pada tab “*Keypairs*”, pilih icon “*Create a new keypairs*”. Berikan nama pada *keypair* yang baru dibuat, kemudian pilih lokasi untuk menyimpan file *keypair* tersebut. *Keypair* ini digunakan untuk melakukan koneksi SSH.



Gambar 5.16 *Key Pairs*

**5.1.2.5. Membuat *Security Groups***

Klik pada tab “*Security Groups*”, pilih grup yang telah ada atau membuat baru dengan klik icon “*Create Groups*” dan berikan nama grup tersebut. Selanjutnya pindah ke bagian “Group Permisions” dan tambahkan *group rules* sesuai dengan kebutuhan untuk akses *cloud* nantinya. Secara umum *port* yang diakses adalah SSH/RDP, jadi kedua port ini harus ada.

Gambar 5.17 *Security Group*

**5.1.2.6. Menjalankan *Images***

Untuk menjalankan *image* dari *hybridfox* perlu dipastikan bahwa *image* telah tersedia dalam daftar pada tab “*Images*”. Adapun langkah – langkahnya sebagai berikut :

1. Cek images dari hybridfox

Gambar. 5.18 *Image List in Hybridfox*

1. Jalankan *instance* dari sebuah *images*

Gambar 5.19 Menjalankan *Instance* dari Sebuah *Images*

1. Apabila jalan akan tampil seperti berikut



Gambar 5.20 *Instance* dengan Status *Running*

1. Tampilan detail dari instance

Gambar 5.21 Informasi Detail *Instance*

1. Selanjutnya klik kanan pada *image storage* yang dipilih (kode *emi*-xxxxx), pilih “*Launch Image*”.

Gambar 5.22 *Connect to Instance*

1. Setelah status *instance* menjadi “*running*”, klik kanan pada *instance* lalu pilih “*Show Console Output*” untuk melihat proses dari *instance*.

Gambar 5.33 *Console Output*

1. Jika hasil dari “*Show Console Output*” seperti diatas, *instance* berjalan dengan baik. Klik “OK” dan lanjutkan dengan klik kanan pada *instance*, pilih “*Connect to Public DNS Name*” untuk me-*remote instance*. Atau dengan melalui *terminal*, kemudian ketikkan :

cd ~/.euca

sudo ssh -i <key pair name> ubuntu@<Instance IP Address/Elastic

1. Apabila koneksi SSH berhasil, lanjutkan dengan mengetik “*yes*”, dan sistem akan melanjutkan untuk *login* ke *instance* yang telah *running*.

Gambar 5.34 Autentikasi dari Host yang dituju

1. Tampilan setelah *user*/*client* berhasil terkoneksi dengan *instance* yang telah dibuat. Sistem operasi yang ditampilkan merupakan pilihan *user* saat membuat/menjalankan *instance*. Sistem tersebut dapat digunakan sesuai kebutuhan user, misalnya sebagai *server*.

Gambar 5.35 *Remote Instance*

1. Setelah *instance* dijalankan menggunakan salah satu *bundle image* atau paket yang disediakan, maka apabila dilihat perbedaan sebelum dan sesudah penggunaan adalah sebagai berikut.

Gambar 5.36 Melihat Ketersediaan Paket *Instance*

Gambar tersebut menunjukkan adanya pengurangan terhadap jumlah ketersediaan *instance* yang dapat dibentuk. Jumlah ketersediaan berkurang satu, karena *instance* yang dibuat sebelumnya menggunakan paket *processor* satu *core*.

**5.1.3. Pembuatan Image pada Server Sistem Pasca**

Image yang akan dibuat adalah dengan mengunakan sistem operasi Ubuntu *Server* 10.10 dengan aplikasi tambahan Apache 2, MySQL *server* 5, PHP 5, PhpMyAdmin untuk penyediaan *web server*. Untuk pembangunan *image* perlu disediakan *disk space* sebesar dua GB dan *memory* sebesar 256 MB di PCAdmin untuk tempat penginstallan Ubuntu *Server* 10.10 dan aplikasi tambahannya.Setelah selesai dibangun maka *image* ubuntu *server* 10.10, kernel dan *ramdisk* diunggah ke *server* satu untuk mendapatkan ID ERI, EKI dan EMI.

**5.1.3.1. Install Apache**

Agar bisa me-manage basis data yang dapat diaskes oleh pengguna sever cloud kita perlu menginstall Apache 2, adapun langkah – langkahnya sebagai berikut.

1. Pertama install webserver apache

# apt-get install apache

Gambar 5.37 Proses Install Apahce2

1. Kemudian install php untuk apache

# apt-get install phpmyadmin

Gambar 5.38 *Install phpmyadmin*

1. Selelah itu kita harus merestart apache terserbut

# /etc/init.d/apache2 restart

1. Buka dengan aplikasi browser

<http://localhost/phpmyadmin>

**5.1.3.2 Upload Sisten Pasca ke Sistem Cloud**

Agar sistem sistem dapat dijalankan pada server cloud yang kita buat maka kita harus mengupload sistem pasca tersebut ke sitem cloud yang kita bangun. Adapun langkah – langkahnya sebaga berikut.

1. Ambil konten wabsite dari sistem pasca

Mengambil konten sistem pasca kita menggunakan aplikasi winSCP

1. Memindahkan server konfensional ke server cloud

Setelah berhasil maka kita dapat melihat hasil sistem yang kita upload ke sistem server cloud.

Gambar 5.39. Tampilan Wab Setelah masuk ke server cloud

**5.2. Analisis Dan Hasil Eksperimen**

Pada tahap ini kita melakukan *review* tahapan-tahapan yang telah berakhir dan mempelajari kriteria dalam prinsip pembelajaran.

**5.2.1. Sebelum dan Sesudah Implementasi Sistem**

|  |  |
| --- | --- |
| Sebelum Sistem Diimplementasikan | Sesudah Sistem Diimplementasikan |
| * Sistem harus memiliki procesoor , hardisk dan memory yang memadai atau tinggi. | * Sistem tidak harus memiliki procesoor , hardisk dan memory yang memadai atau tidak terlalu tinggi. |
| * Ada batasan dengan berapa jumlah "hosting" server yang berjalan di belakangnya | * Tidak ada batasan dengan berapa jumlah "hosting" server yang berjalan di belakangnya. |
| * risiko yang secara konvensional melekat dalam pembelian server fisik dan aplikasi-aplikasi yang ada di dalamnya menjadi resiko Universitas | * skalabilitas dan risiko pembelian infrastruktur yang terlalu banyak atau terlalu sedikit akan menjadi resiko resources cloud computing. |

**5.2.2. Evaluasi Sistem**

Setelah melakukan perancangan dan dicoba oleh penulis berdasarkan penelitian yang dilakukan terdapat beberapa kelebihan dan kekurangan yang ada, yaitu:

1. Kelebihan

* Lebih mudah untuk mengontrol server yang ada.
* Sistem *Cloud Computing menggunaan Eucalyptus Ubuntu Enterprise Cloud* sangat mudah untuk dikonfigurasi karena bisa diatur melalui web browser
* Banyak fitur yang bisa dilakukan seperti hypervisor yang berbayar lainnya seperti migration, backup, dsb

1. Kekurangan

* Membutuhkan koneksi internet yang stabil.
* Mebuthkan spesifikasi hardware yang cukup tinggi pada saat melakukan uji coba.

**5.2.3. Penggunaan Sistem**

Setelah melakukan eksperimen penggunaan system cloud computing menggunakan Eucalyptus UEC pada system informasi pasca sarjana, sistem server cloud dapat berjalan dengan baik.

**BAB VI**

**SIMPULAN DAN SARAN**

## 6.1 Simpulan

Pada proses uji coba implementasi sistem *cloud computing* menggunakan *Eucalyptus – Ubuntu Enterprise Cloud* pada sistem informasi pasca sarjana Universitas Binadarma dapat disimpulkan.

1. *Eucalyptus* merupakan salah satu *framework* perangkat lunak berbasis *Open Source* yang mendukung perkembangan *Cloud Computing* baik *private* maupun *public*, dengan mengimplementasikan *Infrastructure As Service* atau IaaS. Infrastruktur yang dimaksud berupa sebuah sistem operasi yang diperuntukan dalam kebutuhan baik *server* jaringan maupun penggunaan *user* biasa.
2. Sumber daya untuk kebutuhan hidup setiap *instances* secara minimal adalah satu *core* dari *processor* yang digunakan oleh *server node controller*. Jadi banyaknya *instances* tergantung dari banyaknya *node controller* dan jumlah *core* dari setiap *node controller*. Dengan demikian sebuah perangkat *node* dapat disediakan untuk beberapa pengguna sehingga mengurangi kebutuhan *hardware* yang dapat memakan ruangan.
3. *Private cloud* yang dibangun menggunakan *eucalyptus ubuntu enterprise cloud* dapat digunakan sebagai ujicoba/*study* pengembangan sistem dengan perangkat yang terjangkau.
4. Kekurangan jika teknologi *private cloud computing* diterapkan di UBD ialah UBD membutuhkan dana tambahan untuk membangun infrastruktur baru dan pengadaan server yang kompatibel.

## 6.2 Saran

Untuk pengembangan selanjutnya, beberapa saran yang dapat diberikan adalah sebagai berikut :

1. Implementasi hendaknya menggunakan perangkat dan fasilitas yang sesuai dan mendukung.
2. Penggunaan topologi harus sesuai dengan kemampuan perangkat dan kebutuhan pemakaian.
3. Ada baiknya di UBD diadakan penelitian lebih lanjut lagi dan pengadaan hardware untuk pengembangan cloud computing sebagai sarana pendukung mahasiswa. Mulai dari cloud software as a service, platform as a service hingga infrastructure as a service dengan harapan meningkatkan penggunaan dan layanan online yang sudah ada di UBD seperti forum online, e-*learning*, mail server dan layanan lokal UBD yang lainnya. Dengan menggunakan aplikasi *opensource* biaya operasional sistem akan sangat minim dan mudah untuk dikembangkan. UBD memiliki sumberdaya manusia yang sangat mendukung untuk pengembangan dan operasionalnya, sayang jika potensi yang ada tidak dikembangkan.
4. Infrastruktur *private cloud* agar digunakan dan diuji coba sebagai *server*/satu kesatuan sebuah sistem seperti *web server*, *database*, *mail server* dan lain sebagainya.