

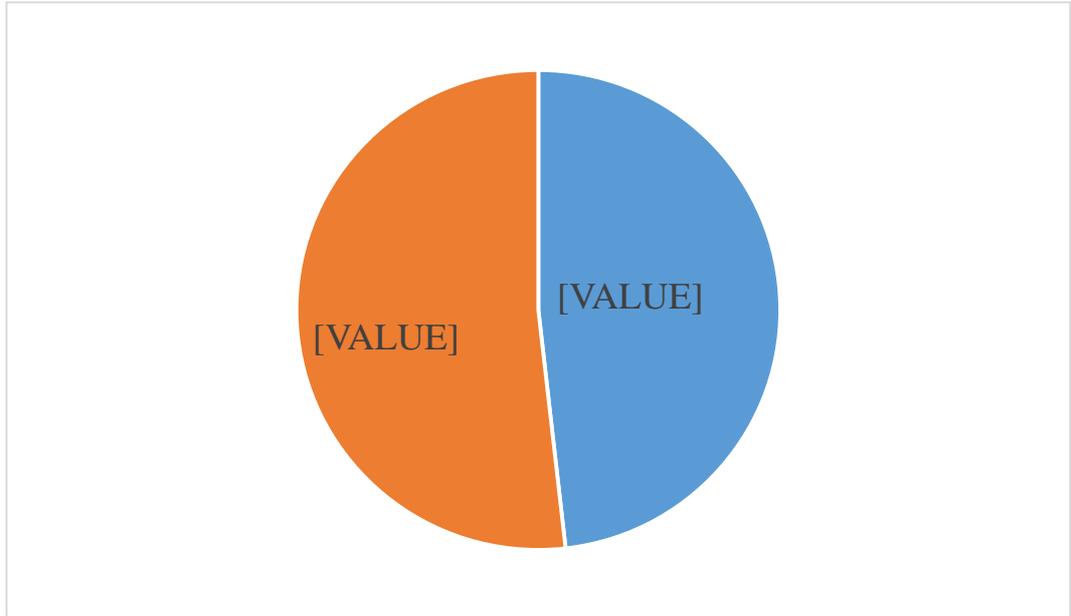
# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Seiring dengan perkembangan teknologi dan informasi seperti saat ini, teknologi informasi mengalami perkembangan yang efisien dengan kemudahan akses, sehingga dalam kegiatan sehari-hari yang tidak mungkin dapat dikerjakan dalam waktu singkat, pada saat ini dapat dilakukan. Pengembangan teknologi komputasi berbasis internet pada masa ini diarahkan pada proses aplikasi sistem yang mudah dan tidak membutuhkan banyak waktu atau tenaga.

Teknologi *cloud computing* atau dikenal dengan komputasi awan, sudah digunakan dan bukan merupakan hal asing lagi. Teknologi *cloud computing* dapat menghemat pengeluaran dibandingkan dengan membangun sendiri infrastruktur jaringan dalam jangka pendek. Kebutuhan untuk teknologi *cloud computing* sebagian besar terdapat pada biaya koneksi dan data *processing* yang sesuai dengan kebutuhan. Pada beberapa perusahaan atau institusi, penggunaan media penyimpanan data dan keamanan menjadi faktor utama yang harus diperhatikan. Biaya yang dikeluarkan untuk mengembangkan media penyimpanan data dan keamanan sangat besar. Inovasi-inovasi yang signifikan dalam hal virtualisasi dan distribusi *computing* termasuk juga peningkatan akses ke internet dengan berkecepatan tinggi telah meningkatkan minat masyarakat kepada *cloud computing*. Hal ini dapat dilihat dari akses internet yang ada di Indonesia pada tahun 2016 sebanyak 132,7 juta *user* dari total jumlah penduduk Indonesia sebesar 256,2 juta orang [8].



**Gambar 1.1** *Penetration Users Internet Indonesia* [8].

Dalam memanfaatkan perkembangan era teknologi digital dan informasi, *cloud computing* merupakan salah satu alternatif bagi perusahaan untuk menekan biaya yang berhubungan dengan penyimpanan data. *Cloud computing* adalah sebuah model komputasi dimana sumber daya, seperti *processor*, *storage*, *network* dan *software* menjadi abstrak dengan diberikannya sebagai layanan di jaringan internet dalam penggunaan pola akses *remote*. *Cloud* sendiri berasal dari simbol yang lazim digunakan untuk melambangkan teknologi internet yang dipakai secara bersama. Secara sederhana, layanan teknologi informasi yang bisa dimanfaatkan atau diakses oleh pelanggan melalui jaringan internet. Seiring berkembangnya teknologi *open-source*, maka muncullah *toolkit* yang memungkinkan seseorang atau suatu organisasi untuk membangun *platform cloud computing*. Proxmox adalah salah satu contoh *open source* pada *cloud computing toolkit* saat ini.

Proxmox merupakan sebuah *platform* perangkat lunak yang mengelola infrastruktur dengan menerapkan gaya IaaS yang terukur pada sifat *privat server*. *Platform proxmox* menjadi sebuah solusi manajemen virtualisasi *server open source* berbasis Qemu/KVM dan LXC yang lengkap, ringan dan mudah. Proxmox menawarkan kemampuan untuk mengelola *server virtual* (VPS) dengan teknologi Linux OpenVZ dan teknologi KVM. Proxmox termasuk distro Linux virtualisasi

berbasis Debian (64 bit), dengan KVM kita tidak hanya bisa menginstal Linux saja akan tetapi, *operating system* Windows pun bisa kita instal, namun, yang membuat istimewa dari proxmox adalah kemudahan dalam instalasi dan administrasi berbasis *web*. Dengan menginstal proxmox, maka *cloud* kita dapat memiliki kemampuan *live migration* dari *virtual machine* dengan melalui *interface web*. Untuk mengatur migrasi satu mesin fisik ke mesin yang lain, maka dibutuhkan minimal dua proxmox.

*Live migration* merupakan kemampuan untuk memindahkan mesin virtual yang sedang berjalan kepada *host* fisik lain tanpa mengurangi interupsi kepada layanan *hypervisor*. *Live migration* transparan kepada pengguna akhir, mesin virtual tetap menyala, koneksi jaringan tetap aktif, dan aplikasi pengguna tetap berjalan saat mesin virtual dipindahkan ke *host* fisik baru. *Live migration* juga mendukung penyimpanan status mesin virtual pada saat itu dan dimulai lagi pada lain waktu. Pada *live migration* memungkinkan konten layanan tetap dapat diakses selama proses *live migration* terjadi. Proses perpindahan memori ini terjadi dalam 3 tahap, yaitu tahap pertama *push phase*, pada tahap ini *source virtual machine* tetap berjalan ketika membuat sebuah *pages* untuk dikirim ke *host destination and upadate* yang terjadi pada *source* tidak dapat lagi ditangani. Tahap kedua, yaitu *stop and copy phase* dalam tahap ini *source virtual machine* berhenti beroperasi, semua telah di *copy* ke *destination host* dan virtual mesin baru dapat berjalan dan pada tahap terakhir, yaitu *pull phase* yang dimana, virtual mesin baru telah dapat di eksekusi dan *update* untuk virtual mesin ini dapat langsung berjalan. *Platform live migration* telah mengizinkan *administrator* untuk memindahkan virtual mesin yang sedang berjalan ke fisikal *host* yang baru. Ini merupakan keuntungan untuk para *service provider* untuk menyediakan *high availability* pada aplikasi lainnya.

Ada tingkatan layanan, dimana sebuah *service provider* berkomitmen kepada pengguna dalam hal penyewaan dengan menjalankan aplikasi yang dideskripsikan dalam SLA (*Service Level Agreement*) yang berhubungan langsung dengan *high availability* dari aplikasi. Berkaitan dengan bisnis, aktifitas seperti *restarting* sebuah mesin untuk perawatan *hardware*, sehingga menghentikan layanan aplikasi adalah suatu hal yang sangat dilarang di zaman sekarang ini. *Live*

*migration* memudahkan penyelesaian masalah itu dengan mengizinkan *administrator* untuk dapat memindahkan virtual mesin dengan sedikit interupsi. Hal ini, membuat dapat terjadinya perawatan fisik *hardware* secara reguler dengan dukungan dinamik rekonfigurasi dan mendukung pemindahan beban komputasi serta mendinginkan pusat data. Dalam melakukan *live migration* di *cloud computing* hal-hal yang perlu diperhatikan adalah kualitas *throughput*, *delay*, *threshold*, ukuran data transfer pada VM serta beban CPU, lama waktu migrasi, dan *downtime* [8].

## 1.2 Rumusan Masalah

Berdasarkan latar belakang masalah di atas, maka rumusan masalah di dalam penelitian ini, yaitu: Bagaimana cara mengoptimalkan *live migration* dari sudut pandang ukuran data transfer, lama waktu migrasi, dan *downtime* dengan menggunakan metode *pre-copy*?

## 1.3 Tujuan

Tujuan dari penelitian ini adalah mengimplementasikan suatu sistem *cloud* yang memiliki kemampuan untuk *live migration* dengan ukuran data transfer, lama waktu migrasi dan *downtime* pada setiap virtual mesin dengan metode *pre-copy*.

## 1.4 Batasan Masalah

Batasan masalah yang digunakan di dalam pelaksanaan tugas akhir kali ini, antara lain:

1. *Cloud computing* dibangun dengan menggunakan sistem operasi Debian 8.8
2. *Cloud computing* dibangun berbasis Proxmox VE 4.4
3. Tidak membahas tentang keamanan sistem
4. *Cloud computing* yang dibangun akan menggunakan algoritma *pre-copy* pada pengaplikasian *live migration*
5. *Cloud* yang di gunakan kali ini sebatas *private cloud*

## **1.5 Metodologi Penelitian**

Metodologi penelitian yang dilakukan dalam penyelesaian tugas akhir ini, yaitu:

1. Tahap Studi literatur, yaitu mempelajari literatur-literatur yang sesuai dengan permasalahan.
2. Membahas penelitian terkait *live migration*, proxmox, dan *cloud computing*.
3. Analisis terhadap *downtime*, lama waktu migrasi, dan ukuran data transfer.
4. Perancangan *live migration* di *server cloud* berbasis proxmox.
5. Tahap analisis hasil dan penelitian kesimpulan.
6. Pembuatan laporan dari hasil penelitian.

## **1.6 Sistematika Penulisan Tugas Akhir**

Pada penelitian ini, secara keseluruhan dapat disusun secara struktural sebagai berikut:

### **BAB 1 PENDAHULUAN**

Berisi tentang latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penelitian.

### **BAB II DASAR TEORI**

Berisi tentang penjelasan mengenai *cloud computing*, proxmox, *hypervisor*, algoritma *pre-copy*, *kernel based virtual machine*, ukuran data transfer, *downtime*, *migration time*, dan *live migration*.

### **BAB III PERANCANGAN SISTEM**

Berisi tentang penjelasan dari gambaran umum sistem, dan serta Perancangan *live migration* di *server cloud* berbasis proxmox dengan metode *pre-copy*.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Berisi tentang pengimplementasian proxmox pada *server* fisik merek xeon yang sebagai *node controller* pertama dan pada PC sebagai *node controller* kedua, dan

pengujian *downtime*, lama waktu migrasi, dan ukuran data transfer dan serta menganalisis hasil pada pengujian.

## **BAB V KESIMPULAN DAN SARAN**

Berisi tentang kesimpulan dari hasil penelitian yang dilakukan dan di rekomendasikan untuk penelitian berikutnya.