

Tugas Individu Piramida Komputasi Awan

Edgrant Henderson Suryajaya
2206025016

I. SEJARAH

Pada tahun 1990-an, perusahaan telekomunikasi menyediakan layanan jaringan komputer yang disebut dengan Virtual Private Network (VPN) yang memungkinkan banyak user menggunakan hardware yang sama. Ini merupakan awal dari konsep cloud computing. Lalu, konsep seperti grid computing, utility computing, dan Software as a Service (SaaS) muncul sebagai cikal bakal dari cloud computing.

- *Grid computing*: Masalah komputasi dibagi menjadi beberapa bagian dan dikerjakan oleh beberapa komputer yang berbeda
- *Utility computing*: menjual layanan komputasi
- *Software as a Service (SaaS)*: menyediakan *software* pada user melalui internet

“Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

Itulah definisi *Cloud Computing* oleh NIST. Dalam kata lain, cloud computing memberikan layanan sumber daya komputasi *sharing* yang mudah dikonfigurasi.

II. ROOTS OF CLOUD COMPUTING

Menurut NIST (*National Institute of Standards and Technology*), *Cloud Computing* memiliki 5 karakteristik utama, 3 model layanan, dan 5 model implementasi.

A. Model Layanan Cloud Computing

- *Software as a Service (SaaS)*: menyediakan aplikasi yang sudah jadi. Pelanggan hanya perlu menggunakan aplikasi tersebut tanpa perlu mengurus infrastruktur atau pembuatan aplikasi. (Contoh: Google Docs, Microsoft Office 365)
- *Platform as a Service (PaaS)*: menyediakan platform untuk pengembangan aplikasi. Pelanggan dapat membuat aplikasi mereka sendiri tanpa perlu mengurus infrastruktur. (Contoh: Vercel, Heroku, Wordpress)
- *Infrastructure as a Service (IaaS)*: menyediakan infrastruktur komputasi, seperti server, jaringan, dan penyimpanan. Pelanggan dapat mengelola infrastruktur tersebut sesuai kebutuhan mereka. (Contoh: Amazon Web Services, Microsoft Azure)

SaaS adalah layanan yang *High Level*, user hanya perlu menggunakan aplikasi dan tidak perlu memikirkan implementasi dibaliknya. PaaS adalah layanan yang *Middle Level*,

user dapat mengembangkan aplikasi mereka sendiri tanpa perlu memikirkan infrastruktur. IaaS adalah layanan yang *Low Level*, user dapat mengelola infrastruktur sesuai kebutuhan mereka.

B. Model Implementasi Cloud Computing

- *Public Cloud*: layanan cloud yang disediakan oleh penyedia layanan cloud dan dapat diakses oleh publik. (Contoh: AWS, Azure, Google Cloud)
- *Private Cloud*: layanan cloud yang disediakan oleh perusahaan sendiri untuk kebutuhan internal biasanya digunakan oleh perusahaan besar.
- *Hybrid Cloud*: gabungan dari public dan private cloud. Perusahaan menggunakan private cloud untuk data sensitif dan public cloud untuk data yang tidak sensitif.
- *Community Cloud*: layanan cloud yang digunakan oleh beberapa organisasi yang memiliki kepentingan yang sama. Contohnya kluster perumahan besar yang menggunakan layanan cloud khusus untuk kluster tersebut.
- *Virtual Private Cloud*: layanan cloud yang mensimulasikan private cloud, tetapi sebenarnya menggunakan public cloud. Contohnya sebuah organisasi memberikan layanan cloud khusus untuk karyawan mereka tetapi sebenarnya data di-host pada AWS.

Model implementasi cloud yang paling sering digunakan pada zaman sekarang adalah *Public Cloud* dan *Hybrid Cloud*. *Public Cloud* digunakan oleh banyak perusahaan karena lebih murah dan mudah digunakan. *Hybrid Cloud* digunakan oleh perusahaan besar yang memiliki data sensitif, tetapi juga ingin menggunakan layanan cloud yang murah.

C. Characteristics of Cloud Computing

- *Rapid Elasticity*: kemampuan untuk menambah atau mengurangi sumber daya komputasi sesuai kebutuhan.
- *Measured Service*: penggunaan sumber daya komputasi diukur dan dilaporkan. Pengguna hanya membayar sumber daya yang digunakan.
- *On-demand Self-service*: sumber daya komputasi dapat digunakan tanpa perlu interaksi dengan penyedia layanan.
- *Ubiquitous Network Access*: sumber daya komputasi dapat diakses dari mana saja dan kapan saja.
- *Resource Pooling*: sumber daya komputasi disediakan secara bersamaan untuk beberapa pengguna. Sumber daya ini dapat dialokasikan sesuai kebutuhan.

III. VIRTUALISASI

Tidak ada aplikasi yang menggunakan 100% sumber daya komputasi yang ada pada server setiap saat, sehingga penyedia layanan cloud menggunakan teknologi virtualisasi untuk

membagi sumber daya komputasi tersebut ke beberapa user. Virtualisasi memungkinkan satu server fisik digunakan oleh beberapa user secara bersamaan.

Virtualisasi adalah teknologi yang memungkinkan satu server fisik untuk menjalankan beberapa sistem operasi secara bersamaan yang masing-masing berjalan secara independen dan terisolasi. Terdapat beberapa tipe virtualisasi yang digunakan pada cloud, beberapa antaranya adalah *Hypervisor-based Virtualization* dan *Container-based Virtualization*.

A. Hypervisor-based Virtualization

Hypervisor adalah software yang menjalankan beberapa sistem operasi secara bersamaan pada satu server fisik. Terdapat dua tipe *Hypervisor*, yaitu *Type 1 Hypervisor* dan *Type 2 Hypervisor*.

- *Type 1 Hypervisor*: *Hypervisor* berjalan langsung pada hardware server. Contoh dari *Type 1 Hypervisor* adalah VMware ESXi, Microsoft Hyper-V, dan Xen.
- *Type 2 Hypervisor*: *Hypervisor* berjalan di atas sistem operasi yang sudah ada. Contoh dari *Type 2 Hypervisor* adalah VMware Workstation, Oracle VirtualBox, dan Parallels Desktop.

Hypervisor-based Virtualization memungkinkan user untuk menjalankan sistem operasi yang berbeda pada satu server fisik. Setiap sistem operasi yang berjalan di-host oleh *Hypervisor* disebut dengan *Virtual Machine*. *Virtual Machine* memiliki sumber daya komputasi yang terisolasi dari *Virtual Machine* lainnya yang ditentukan saat pembuatan *Virtual Machine* tersebut.

B. Container-based Virtualization

Container-based Virtualization adalah teknologi yang memungkinkan user untuk menjalankan aplikasi dan dependensinya dalam wadah yang terisolasi dari aplikasi lainnya. *Container* berbagi kernel dari sistem operasi yang sama. Contoh dari *Container-based Virtualization* adalah Docker, Kubernetes, dan LXC.

Perbedaan utama antara *Hypervisor-based Virtualization* dan *Container-based Virtualization* adalah pembagaaian sumber daya komputasi. *Hypervisor-based Virtualization* membagi sumber daya komputasi secara fisik seperti RAM, CPU, dan storage *Virtual Machine* dibuat, sedangkan *Container-based Virtualization* dapat membagi sumber daya komputasi secara dinamis sesuai kebutuhan aplikasi.

IV. CONCERN OF CLOUD PLAYERS

Dalam penggunaan layanan cloud, terdapat banyak *stakeholder* yang memiliki kepentingan masing-masing. *Stakeholder* tersebut adalah manajer fasilitas, penyedia layanan, pengguna layanan, manajer IT, dan *End User*. Kepentingan dari *stakeholder* tersebut adalah berbeda-beda

- *Manajer fasilitas*: memastikan infrastruktur *data center* (DC) berjalan dengan baik dan aman. Memaksimalkan performa, meminimalkan biaya operasional.

- *Penyedia layanan*: memastikan layanan cloud yang disediakan aman, handal, dan tidak terputus. Memaksimalkan utilisasi peralatan, merencanakan kapasitas dengan mengkonsiderasi kemungkinan pertumbuhan.
- *Pengguna layanan*: memastikan *End User* dapat menggunakan layanan mereka dengan baik.
- *End User*: mementingkan data mereka aman, privasi terjaga, dan layanan yang mereka gunakan dapat diakses kapan saja.
- *Manajer IT*: IT dari sisi pengguna layanan, memastikan layanan yang digunakan oleh *End User* dapat diakses kapan saja, aman, dan privasi terjaga. Selain itu, merencanakan kemungkinan pertumbuhan layanan.

Kelima *stakeholder* tersebut memiliki kepentingan yang berbeda-beda, tetapi terdapat beberapa kepentingan yang sama, yaitu keamanan data, privasi, dan ketersediaan layanan. Teknologi *Cloud Computing* yang digunakan harus memenuhi kepentingan dari kelima *stakeholder* tersebut.

V. CONSIDERATIONS FOR DATA CENTERS

A. Migration

DC harus mempertimbangkan migrasi data dari satu DC ke DC lainnya. Kemungkinan terjadinya *downtime* harus diminimalkan. DC harus memastikan data yang di-migrate aman, tidak terjadi kehilangan data, dan data dapat diakses kapan saja.

B. Performance

DC harus memastikan performa layanan yang diberikan kepada pengguna layanan. DC harus memastikan sumber daya komputasi yang digunakan oleh pengguna layanan optimal, tidak terjadi *overload* atau *underload* pada sumber daya komputasi. DC harus memastikan layanan yang diberikan dapat diakses kapan saja dengan performa yang baik.

C. Security

Public cloud sering menjadi target serangan *cyber*. Tergantung dengan model layanan yang digunakan, tanggung jawab keamanan data dapat berbeda-beda. Sebagai contoh, pada model *Infrastructure as a Service (IaaS)*, penyedia layanan hanya bertanggung jawab untuk keamanan infrastruktur, sedangkan pengguna layanan bertanggung jawab untuk keamanan data mereka. DC harus memastikan data yang disimpan aman, tidak terjadi kebocoran data, dan data tidak diakses oleh pihak yang tidak berhak. Namun, pada model *Software as a Service (SaaS)* atau *Platform as a Service (PaaS)*, penyedia layanan bertanggung jawab untuk keamanan data dan aplikasi (hanya pada SaaS) yang disimpan oleh pengguna layanan.

REFERENCES

- [1] N. K. Sehgal, V. Chandra Joshi, A. Aggarwal, A. Mathur, and G. S. Tomar, "Chapter 3," in *Cloud Computing with Security*. Springer Nature Switzerland AG, 2020. doi: 10.1007/978-3-030-24612-9_3.
- [2] Amazon Web Services, "What is a hypervisor?," Available: <https://aws.amazon.com/what-is/hypervisor/>. [Accessed: Feb. 20, 2025].
- [3] Amazon Web Services, "The difference between Docker and VMs," Available: <https://aws.amazon.com/id/compare/the-difference-between-docker-vm/>. [Accessed: Feb. 20, 2025].