

PROYEK 8

MEMBANGUN PRIVATE CLOUD CLUSTER DENGAN PROXMOX VE



Diajukan sebagai salah satu tugas mata kuliah Sistem Operasi

Dosen Pengampu: Ferdi Chahyadi, S.Kom., M.C

Oleh: Kelompok BLACKPINK

- | | |
|---------------------------|-----------------|
| 1. Dhini Khairunnisa | NIM. 2401020016 |
| 2. Erlinda Amira Putri S. | NIM. 2401020021 |
| 3. Nikita Arzetty Siregar | NIM. 2401020030 |
| 4. Rismayanti | NIM. 2401020041 |
| 5. Ary Subakti | NIM. 2301020065 |

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN TEKNOLOGI KEMARITIMAN
UNIVERSITAS MARITIM RAJA ALI HAJI
TANJUNGPINANG
2025**

DAFTAR ISI

BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Tujuan	2
1.4 Manfaat	2
BAB II LANDASAN TEORI.....	4
2.1 Sistem Operasi dan Virtualisasi.....	4
2.2 Clustering	4
2.3 Proxmox Virtual Environment (VE).....	4
2.4 Ceph Distributed Storage	5
2.5 High Availability (HA)	5
2.6 Live Migration.....	6
BAB III METODE PENELITIAN	7
3.1 Desain Penelitian	7
3.2 Tahapan Implementasi	7
3.3 Tools dan Software	9
3.4 Parameter Pengujian	9
BAB IV PENUTUP	11
DAFTAR PUSTAKA	12

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di zaman serba digital seperti sekarang, kebutuhan akan infrastruktur IT yang kuat, mudah dikembangkan, dan selalu siap digunakan semakin besar. Cloud computing jadi salah satu solusi utama, terutama dengan konsep clustering yang membuat beberapa server bisa bekerja seperti satu sistem. Private cloud cluster adalah bentuk cloud yang dikelola sendiri di dalam lingkungan organisasi. Dengan model ini, kita punya kontrol penuh terhadap data, keamanan, dan performanya. Berbeda dengan public cloud, private cloud lebih fleksibel untuk dikustomisasi dan biasanya lebih hemat untuk penggunaan jangka panjang.

Proxmox Virtual Environment (VE) adalah platform virtualisasi open-source berbasis Debian Linux yang menggabungkan teknologi KVM untuk virtualisasi penuh dan LXC untuk container. Proxmox VE punya fitur cluster yang cukup kuat, sehingga beberapa server dapat digabung dan bekerja bersama. Platform ini juga mendukung High Availability (HA), live migration, dan penyimpanan terdistribusi. Clustering sendiri adalah teknik menghubungkan beberapa komputer atau server agar menjadi satu sistem yang berbagi tugas. Tujuannya untuk meningkatkan ketersediaan, keandalan, dan kemampuan sistem dalam menangani beban. Dalam dunia sistem operasi, konsep ini penting untuk memahami cara kerja pembagian resource, load balancing, dan toleransi terhadap kegagalan.

Pada project ini, kami akan membangun private cloud cluster dengan dua node Proxmox VE yang menggunakan Ceph sebagai distributed storage. Kami juga akan menerapkan High Availability agar sistem dapat melakukan failover otomatis, serta menguji live migration untuk memastikan proses pemeliharaan dapat berjalan tanpa downtime.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam project ini adalah:

1. Bagaimana cara membangun cluster Proxmox VE multi-node yang terintegrasi?

2. Bagaimana cara mengkonfigurasi Ceph distributed storage pada cluster Proxmox VE?
3. Bagaimana cara mengimplementasikan High Availability untuk virtual machine?
4. Bagaimana cara melakukan live migration virtual machine antar node tanpa downtime?
5. Bagaimana mekanisme failover bekerja ketika salah satu node mengalami kegagalan?
6. Bagaimana performa cluster dalam menangani workload dan resource allocation?

1.3 Tujuan

Tujuan Umum: Membangun dan mengimplementasikan private cloud cluster menggunakan Proxmox VE yang mampu menjalankan High Availability, live migration, dan distributed storage dengan performa optimal.

Tujuan Khusus:

1. Menginstal dan mengkonfigurasi Proxmox VE pada minimal 2 node server
2. Membentuk cluster Proxmox VE dengan mekanisme quorum voting
3. Mengimplementasikan Ceph distributed storage sebagai shared storage untuk cluster
4. Mengkonfigurasi High Availability (HA) pada virtual machine
5. Melakukan live migration virtual machine antar node tanpa downtime
6. Menguji mekanisme failover ketika salah satu node mengalami kegagalan
7. Menganalisis performa cluster dari aspek CPU, memory, network, dan storage I/O
8. Mendokumentasikan seluruh proses implementasi, konfigurasi, dan hasil pengujian

1.4 Manfaat

Manfaat Akademis:

1. Memahami konsep clustering dan distributed computing dalam sistem operasi
2. Menguasai teknologi virtualisasi berbasis KVM dan containerization dengan LXC
3. Memahami mekanisme High Availability dan fault tolerance
4. Mengimplementasikan teori sistem operasi dalam aplikasi praktis

Manfaat Praktis:

1. Memiliki Kompetensi dalam membangun infrastruktur cloud private

2. Menguasai skill yang dibutuhkan industri IT untuk posisi DevOps Engineer dan System Administrator
3. Memiliki pengalaman hands-on dengan teknologi enterprise-level
4. Membangun portofolio teknis untuk pengembangan karir

Manfaat untuk Institusi:

1. Menghasilkan dokumentasi lengkap yang dapat menjadi refensi untuk pembelajaran
2. Memberikan kontribusi dalam pengembangan kurikulum praktikum sistem operasi
3. Meningkatkan kualitas pembelajaran berbasis project

BAB II

LANDASAN TEORI

2.1 Sistem Operasi dan Virtualisasi

Sistem operasi adalah software yang mengatur kerja hardware komputer dan menyediakan layanan untuk berbagai aplikasi. Di era sekarang, teknologi virtualisasi memungkinkan satu sistem operasi utama (host) menjalankan beberapa sistem operasi lain (guest) secara bersamaan. Salah satu teknologi virtualisasi adalah KVM (Kernel-based Virtual Machine), yang memanfaatkan kernel Linux agar bisa berfungsi sebagai hypervisor type-1. Dengan cara ini, performa virtual machine bisa mendekati performa sistem asli (bare-metal).

2.2 Clustering

Clustering adalah teknik menghubungkan beberapa komputer (node) supaya bekerja sebagai satu kesatuan sistem. Tujuan utamanya antara lain:

1. **High Availability (HA):** memastikan layanan tetap aktif meskipun ada perangkat yang rusak.
2. **Load Balancing:** membagi beban kerja agar tidak menumpuk di satu server.
3. **Scalability:** kapasitas bisa diperbesar cukup dengan menambah node baru.
4. **Fault Tolerance:** sistem tetap bisa berjalan walaupun ada bagian yang mengalami kegagalan.

Dalam sebuah cluster, ada mekanisme yang disebut **quorum** untuk mencegah terjadinya split-brain, yaitu situasi ketika cluster terbelah menjadi beberapa bagian dan masing-masing menganggap dirinya sebagai pemilik kendali utama. Mekanisme quorum inilah yang menjaga cluster tetap konsisten dan tidak terjadi konflik.

2.3 Proxmox Virtual Environment (VE)

Proxmox VE adalah platform virtualisasi open-source yang menyediakan dua jenis virtualisasi sekaligus:

- **KVM**, untuk full virtualization.
- **LXC**, untuk virtualisasi level sistem operasi (container).

Proxmox juga mendukung fitur clustering yang menggunakan **Corosync** sebagai media komunikasi antar-node sekaligus pengatur quorum. Untuk penyimpanan, Proxmox bisa memakai berbagai jenis storage seperti Ceph, ZFS, atau NFS. Semua fitur tersebut dapat dikelola dengan mudah melalui antarmuka web yang sudah disediakan.

2.4 Ceph Distributed Storage

Ceph adalah sistem penyimpanan terdistribusi yang dirancang untuk bisa terus berkembang dan tetap andal meskipun jumlah node bertambah banyak. Ceph memiliki beberapa keunggulan:

1. **Scalable**, bisa dikembangkan dari ukuran kecil hingga sangat besar.
2. **Reliable**, karena data direplikasi otomatis sebagai cadangan.
3. **Self-healing**, dapat melakukan pemulihan otomatis jika ada node yang gagal.
4. **No Single Point of Failure**, sistem tetap berjalan meskipun salah satu komponen bermasalah.

Komponen penting dalam Ceph meliputi:

1. **OSD (Object Storage Daemon)**: Menyimpan data dan menangani proses replikasi.
2. **Monitor**: Mengelola peta cluster dan memantau status keseluruhan.
3. **Manager**: Menyediakan fungsi monitoring tambahan dan pengelolaan cluster.
4. **MDS (Metadata Server)**: Dipakai jika menggunakan CephFS, namun tidak dipakai dalam project ini.

2.5 High Availability (HA)

High Availability berarti sistem tetap bisa beroperasi dengan downtime yang sangat kecil. Di Proxmox, HA dicapai melalui beberapa mekanisme:

1. **Resource Manager**, yang memantau status VM atau container.
2. **Fencing**, untuk memastikan node yang sedang bermasalah tidak mengganggu cluster.
3. **Automatic Failover**, yaitu proses pemindahan VM secara otomatis ke node lain jika node utamanya gagal.
4. **Priority Groups**, digunakan untuk menentukan VM mana yang harus dipulihkan lebih dulu.

2.6 Live Migration

Live migration adalah teknik memindahkan virtual machine yang sedang aktif dari satu node ke node lain tanpa menghentikan layanan. Prosesnya terdiri dari:

1. **Pre-copy**, yaitu menyalin sebagian besar data memori VM ke node tujuan.
2. **Stop-and-copy**, menghentikan VM sebentar dan menyalin sisa memorinya.
3. **Resume**, menjalankan VM kembali di node tujuan.

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Penelitian ini menggunakan metode eksperimen dengan pendekatan langsung di lapangan (praktis). Setiap tahap dikerjakan secara berurutan, mulai dari instalasi dasar sistem sampai ke proses pengujian fitur-fitur yang lebih kompleks. Tujuannya agar hasil implementasi dapat diamati secara nyata dan dapat diuji performanya.

3.2 Tahapan Implementasi

Minggu 12: Instalasi dan Konfigurasi Dasar

Kegiatan:

Pada tahap awal, dilakukan persiapan lingkungan kerja menggunakan server fisik atau virtual machine. Setelah itu, Proxmox VE diinstal pada tiga node, dilanjutkan dengan pengaturan hostname, konfigurasi jaringan, dan update sistem. Langkah terakhir adalah memastikan semua node bisa saling terhubung tanpa masalah.

Output:

1. Tiga node Proxmox VE terpasang dan berfungsi normal
2. Konfigurasi jaringan tersetting dengan benar
3. Dokumentasi proses instalasi selesai dibuat

Minggu 13: Pembentukan Cluster dan Ceph Storage

Kegiatan:

Tahap berikutnya adalah membangun cluster. Node pertama dibuat sebagai cluster utama, kemudian node kedua dan ketiga ditambahkan. Setelah cluster terbentuk dan quorum terverifikasi, proses instalasi Ceph dilakukan. Semua node diberi peran sebagai monitor, OSD dibuat untuk menyimpan data, dan pool dengan replikasi tiga kali dibentuk. Storage ini kemudian disambungkan langsung dengan Proxmox.

Output:

1. Cluster Proxmox VE lengkap dengan tiga node
2. Ceph berjalan stabil dengan status HEALTH_OK
3. Storage pool siap dipakai untuk virtual machine

Minggu 14: High Availability dan Pengujian

Kegiatan:

Pada tahap ini dibuat sebuah VM untuk pengujian. Fitur High Availability diaktifkan, lalu dilakukan uji live migration antar node, serta uji failover dengan mematikan salah satu node. Selain itu, dilakukan pemantauan resource seperti CPU, RAM, network, dan storage. Pengujian backup–restore, stress testing, serta troubleshooting juga dilakukan untuk melihat stabilitas sistem.

Output:

1. VM dengan High Availability berjalan normal
2. Live migration dapat dilakukan tanpa downtime
3. Failover terbukti bekerja
4. Tersedia data hasil monitoring dan benchmark
5. Log sistem dan grafik performa terdokumentasi

Minggu 15: Finalisasi dan Dokumentasi

Kegiatan:

Tahap terakhir meliputi penyempurnaan konfigurasi, pembuatan dokumentasi lengkap, penyusunan diagram arsitektur, serta analisis hasil pengujian. Semua konfigurasi dikumpulkan dalam repository GitHub. Selain itu, dilakukan persiapan untuk demo dan presentasi akhir.

Output:

1. Laporan akhir tersusun lengkap
2. Repository GitHub tersedia untuk publik
3. Materi presentasi dan demo sistem siap digunakan

3.3 Tools dan Software

Software Utama:

1. Proxmox VE 8.x
2. Ceph Quincy/Reef (versi yang terintegrasi dengan Proxmox)
3. Ubuntu Server 22.04 LTS sebagai sistem operasi untuk VM pengujian

Tools Pengujian:

1. stress-ng untuk uji beban CPU dan memori
2. sysbench untuk benchmarking performa
3. iperf3 untuk menguji kecepatan jaringan
4. fio untuk pengujian I/O penyimpanan

Tools Monitoring:

1. Monitoring bawaan Proxmox
2. Dashboard Ceph
3. htop untuk melihat proses
4. iotop untuk memantau aktivitas I/O

3.4 Parameter Pengujian

Live Migration

1. Lama proses migrasi
2. Durasi downtime (jika ada)
3. Penggunaan resource selama migrasi

Failover

1. Waktu deteksi kegagalan node
2. Lama VM dipindahkan ke node lain
3. Kondisi integritas data setelah failover

Performansi Sistem

1. Performa CPU (MIPS)
2. Kecepatan memori (MB/s)
3. Performa disk (IOPS, read/write)
4. Throughput jaringan (Mbps)

Pengujian Storage

1. Status kesehatan cluster Ceph
2. Latensi replikasi
3. Waktu pemulihan saat salah satu OSD gagal

BAB IV

PENUTUP

Project ini dibuat untuk memberikan pemahaman yang lebih menyeluruh tentang teknologi clustering, virtualisasi, dan penyimpanan terdistribusi yang semuanya sangat dibutuhkan dalam dunia IT saat ini.

Dengan mengikuti metode yang sudah disusun secara bertahap, kami berencana menyelesaikan project ini sesuai jadwal dan menghasilkan output yang sesuai dengan standar penilaian mata kuliah Sistem Operasi.

Harapannya, project ini tidak hanya memberikan nilai akademis tetapi juga menjadi pengalaman langsung yang bermanfaat untuk meningkatkan keterampilan kami dalam bidang infrastruktur dan cloud computing.

DAFTAR PUSTAKA

Tanenbaum, A. S., & Bos, H. (2014). *Modern operating systems* (4th ed.). Pearson Education.

Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). *Operating system concepts* (10th ed.). John Wiley & Sons.

Portnoy, M. (2016). *Virtualization essentials* (2nd ed.). Sybex.

Canonical Ltd. (2024). *Ubuntu Server Guide – Virtualization*. Retrieved from <https://ubuntu.com/server/docs/virtualization-introduction>

Proxmox VE Wiki. (2024). *Proxmox VE Wiki*. Retrieved from <https://pve.proxmox.com/wiki/>

