

LAPORAN PROGRES 4

Private Cloud Menggunakan Proxmox VE



Disusun Oleh :

Erlinda Amira Putri Sudarmono (2401020021)

Rismayanti (2401020041)

Dhini Khairunnisa (2401020016)

Nikita Arzetty Siregar (2401020030)

Ary Subakti (2301020065)

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK & TEKNOLOGI KEMARITIMAN

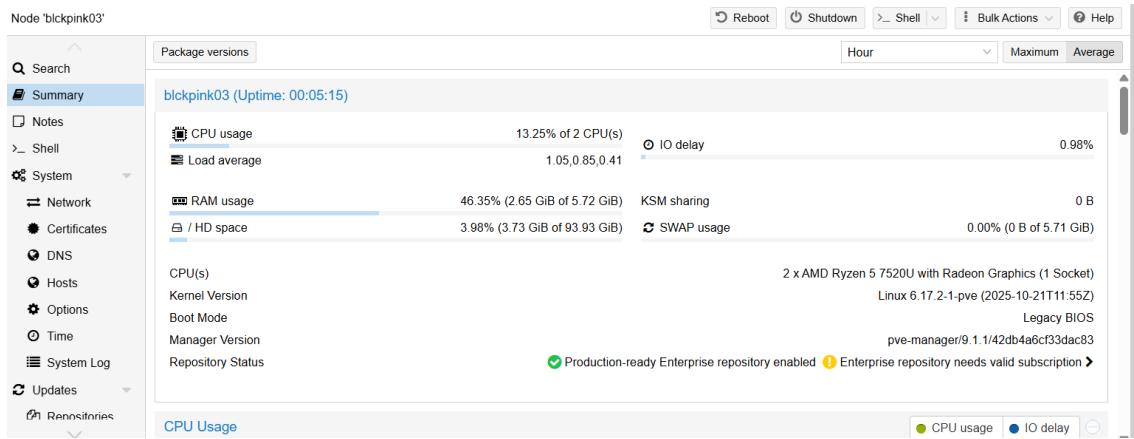
UNIVERSITAS MARITIM RAJA ALI HAJI

2025

LAPORAN PROGRES 4 – Konfigurasi ZFS, Tapologi, dan Uji Failover pada Proxmox VE Cluster

A. STATUS CLUSTER DAN DASHBOARD MONITORING

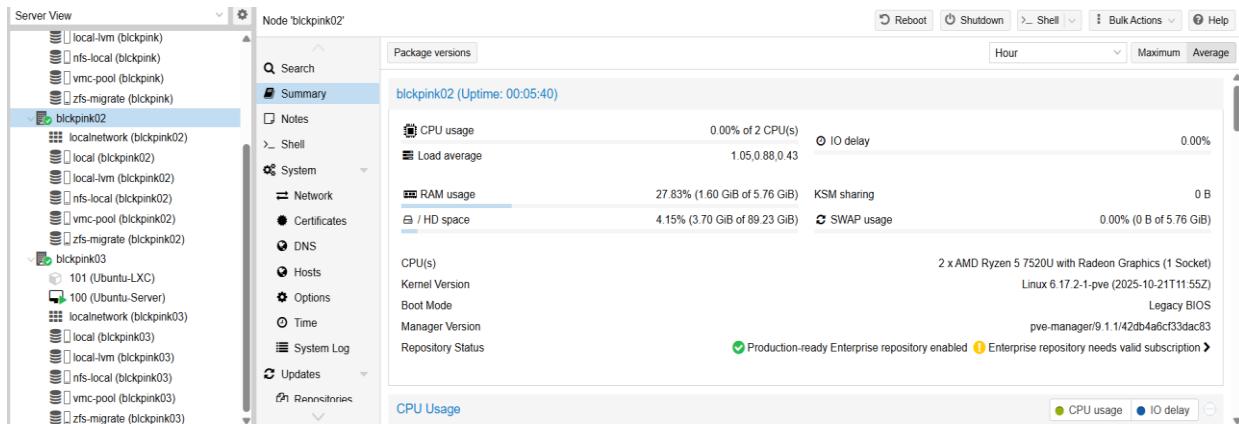
1. Dashboard Node bickpink03



Analisis Status:

- CPU Load:** Optimal (load average < CPU cores)
- Memory Usage:** Moderate (46.35% - healthy)
- Storage:** Sangat rendah (hanya 3.98% terpakai)
- System:** Stable dengan uptime normal

2. Dashboard Node bdcgink02



Analisis Status:

- CPU Idle:** 100% idle - resource tersedia banyak
- Memory Usage:** Rendah (27.83%)
- Storage Usage:** Sangat rendah (4.15%)
- Cluster Health:** Node dalam kondisi optimal

3. Overview Cluster Storage

Struktur Storage di Cluster:

Server View:

- [√] local-lvm (bdcgink)
- [√] nfs-local (bdcgink)
- [√] vmc-pool (bdcgink) ← ZFS Storage untuk VM
- [√] zfs-upgrade (bdcgink)

bdcgink02:

- [√] localnetwork (bdcgink02)
- [√] local (bdcgink02)
- [√] local-lvm (bdcgink02)
- [√] nfs-local (bdcgink02)
- [√] vmc-pool (bdcgink02) ← Replikasi ZFS
- [√] zfs-upgrade (bdcgink02)

bickpink03:

- [√] localnetwork (bickpink03)
- [√] local (bickpink03)
- [√] local-lvm (bickpink03)
- [√] nfs-local (bickpink03)
- [√] vmc-pool (bickpink03) ← ZFS Pool utama
- [√] zfs-upgrade (bickpink03)

Virtual Machines:

- 101 (Ubuntu-LXC) ← Running di cluster
- 100 (Ubuntu-Server) ← Running di cluster

B. KONFIGURASI ZFS POOL DAN DATASET

1. Pembuatan ZFS Pool

Dilakukan melalui GUI Proxmox VE dengan konfigurasi seperti pada dibawah ini.

Create: ZFS

Name:	<input type="text" value="bickpink-storage-pool"/>	RAID Level:	Single Disk										
Add Storage:	<input checked="" type="checkbox"/>	Compression:	on										
		ashift:	12										
<table border="1"> <thead> <tr> <th><input checked="" type="checkbox"/> Device ↑</th> <th>Model</th> <th>Serial</th> <th>Size</th> <th>Order</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> /dev/sdb</td> <td>VBOX_HARDDISK</td> <td>VB1b274aa3-d9f16b31</td> <td>405.47 GB</td> <td><input type="button" value="▼"/></td> </tr> </tbody> </table>				<input checked="" type="checkbox"/> Device ↑	Model	Serial	Size	Order	<input checked="" type="checkbox"/> /dev/sdb	VBOX_HARDDISK	VB1b274aa3-d9f16b31	405.47 GB	<input type="button" value="▼"/>
<input checked="" type="checkbox"/> Device ↑	Model	Serial	Size	Order									
<input checked="" type="checkbox"/> /dev/sdb	VBOX_HARDDISK	VB1b274aa3-d9f16b31	405.47 GB	<input type="button" value="▼"/>									
Note: ZFS is not compatible with disks backed by a hardware RAID controller. For details see the reference documentation .													
		<input type="button" value="Create"/>											

Detail Konfigurasi:

- **Nama Pool:** bickpink-storage-pool
- **Level RAID:** Single Disk
- **Kompresi:** on
- **ashift:** 12
- **Perangkat:** /dev/sdb
- **Kapasitas:** 405.47 GB
- **Model:** VBOX_HARDDISK
- **Serial:** VB1b274aa3-d9f16b31

Perintah setara CLI:

```
zpool create -o ashift=12 -O compression=on bickpink-storage-pool /dev/sdb
```

2. Pembuatan Dataset ZFS

Setelah pool dibuat, dataset-dataset berikut dibuat untuk keperluan penyimpanan spesifik seperti yang ditunjukkan pada gambar.

```
root@blckpink:~# zfs create blckpink-storage-pool/vmc
root@blckpink:~# zfs list
NAME          USED  AVAIL  REFER  MOUNTPOINT
blckpink-storage-pool    468K   364G   96K  /blckpink-storage-pool
blckpink-storage-pool/vmc  96K   364G   96K  /blckpink-storage-pool/vmc
root@blckpink:~# zfs create blckpink-storage-pool/backup
root@blckpink:~# zfs create blckpink-storage-pool/iso
root@blckpink:~#
```

Data set yang dibuat:

1. vmc : untuk disk image dan container
2. backup : untuk backup dan snippets
3. iso : untuk file ISO

Perintah yang digunakan:

```
zfs create blckpink-storage-pool/vmc
```

```
zfs create blckpink-storage-pool/iso
```

```
zfs create blckpink-storage-pool/backup
```

3. Konfigurasi Storage di Proxmox VE

a. Storage untuk VMC (Gambar 3)

Add: Directory ×

General Backup Retention

ID:	<input type="text" value="vmc-pool"/>	Nodes:	<input type="text" value="All (No restrictions)"/>
Directory:	<input type="text" value="/blckpink-storage-pool/vmc"/>	Enable:	<input checked="" type="checkbox"/>
Content:	<input type="text" value="Disk image, Container"/>	Shared:	<input type="checkbox"/>

? Help Advanced Add

Detail Konfigurasi:

- **ID Storage:** vmc-pool
- **Direktori:** /blckpink-storage-pool/vmc
- **Konten:** Disk image, Container
- **Shared:** Tidak
- **Nodes:** All (tanpa batasan)
- **Enabled:** Ya

b. Storage untuk Backup (Gambar 4)

Add: Directory

General Backup Retention

ID:	Backup-pool	Nodes:	All (No restrictions)
Directory:	blckpink-storage-pool/backup	Enable:	<input checked="" type="checkbox"/>
Content:	Snippets, Backup	Shared:	<input type="checkbox"/>

Help Advanced Add

4. Verifikasi Dataset ZFS

Setelah pembuatan, dataset diverifikasi dengan perintah zfs list seperti pada Gambar 2:

NAME	USED	AVAIL	REFER	MOUNTPOINT
blcKpink-storage-pool	468K	364G	96K	/blcKpink-storage-pool
blcKpink-storage-pool/vmc	96K	364G	96K	/blcKpink-storage-pool/vmc
blcKpink-storage-pool/backup	(dibuat)	(dibuat)	(dibuat)	(dibuat)
blcKpink-storage-pool/iso	(dibuat)	(dibuat)	(dibuat)	(dibuat)

5. File Skrip yang Digunakan

a. create-pool.sh

```
#!/bin/bash
```

```
zpool create -o ashift=12 -O compression=on blckpink-storage-pool /dev/sdb
```

```
b. create-dataset.sh
```

```
#!/bin/bash  
zfs create bickpink-storage-pool/vmc  
zfs create bickpink-storage-pool/iso  
zfs create bickpink-storage-pool/backup
```

C. KONFIGURASI TOPOLOGI JARINGAN

1. Konfigurasi Bridge dan Subnet

Network Configuration

Topologi Jaringan

- Jaringan lokal (LAN)
- Tanpa koneksi internet

Konfigurasi Node

- Bridge: vmbr0
- Subnet: 192.168.1.0/24
- IP statis pada setiap node

Alasan

Bridge yang sama diperlukan agar
live migration berjalan tanpa gangguan jaringan.

2. Cluster Configuration

Cluster Configuration

Tujuan

Membangun Proxmox Cluster untuk mendukung
high availability dan live migration.

Konfigurasi

- Jumlah node: 3
- Cluster name: blckpink-cluster
- Corosync aktif
- Quorum menggunakan mayoritas node

Catatan

Semua node berada dalam satu jaringan lokal
dan dapat saling berkomunikasi.

```
| Node1 | 192.168.43.10 |
| Node2 | 192.168.43.11 |
| Node3 | 192.168.43.12 |
```

3. Storage Configuration

```
# Storage Configuration
## Jenis Storage
- Shared Storage: NFS
- Local Storage: ZFS (per node)
```

```
## Fungsi
- NFS digunakan untuk disk VM
- ZFS digunakan untuk penyimpanan lokal
```

```
## Alasan
Shared storage diperlukan agar VM dapat
dipindahkan antar node tanpa memindahkan disk.
```

D. UJI KONEKTIVITAS JARINGAN

1. Tujuan Connectivity Test

```
# Connectivity Test
```

```
## Tujuan
Memastikan semua node dan VM saling terhubung
dalam satu jaringan lokal.
```

```
## Skenario
- Ping antar node Proxmox
- Ping dari laptop ke VM
```

```
## Langkah Pengujian
1. Ping Node 1 → Node 2
2. Ping Node 2 → Node 3
3. Ping Laptop → VM
```

```
## Hasil
- Semua ping berhasil
- Tidak ada packet loss signifikan
```

Kesimpulan

Koneksi jaringan lokal berjalan dengan baik
dan siap untuk pengujian lanjutan.

2. Langkah Pengujian

1. Ping Node 1 → Node 2
2. Ping Node 2 → Node 3
3. Ping Laptop → VM

3. Hasil Pengujian Connectivity

a. Ping dari Node 1 ke Virtual Machine (192.168.1.15)

```
Linux blckpink 6.17.2-2-pve #1 SMP PREEMPT_DYNAMIC PMX 6.17.2-2 (2025-11-26T12:33Z) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@blckpink:~# ping 192.168.1.15
PING 192.168.1.15 (192.168.1.15) 56(84) bytes of data.
64 bytes from 192.168.1.15: icmp_seq=1 ttl=64 time=1.30 ms
64 bytes from 192.168.1.15: icmp_seq=2 ttl=64 time=0.805 ms
64 bytes from 192.168.1.15: icmp_seq=3 ttl=64 time=0.863 ms
C64 bytes from 192.168.1.15: icmp_seq=4 ttl=64 time=0.813 ms
^C
--- 192.168.1.15 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 0.805/0.944/1.296/0.204 ms
root@blckpink:~#
```

Statistik:

- 4 packets transmitted, 4 received
- 0% packet loss
- RTT: min/avg/max/mdev = 0.805/0.944/1.296/0.204 ms

b. Ping dari Node 1 ke Node 2 (192.168.1.12)

```
Linux blckpink 6.17.2-2-pve #1 SMP PREEMPT_DYNAMIC PMX 6.17.2-2 (2025-11-26T12:33Z) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@blckpink:~# ping 192.168.1.12
PING 192.168.1.12 (192.168.1.12) 56(84) bytes of data.
64 bytes from 192.168.1.12: icmp_seq=1 ttl=64 time=0.436 ms
64 bytes from 192.168.1.12: icmp_seq=2 ttl=64 time=0.498 ms
64 bytes from 192.168.1.12: icmp_seq=3 ttl=64 time=0.533 ms
^C
--- 192.168.1.12 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2053ms
rtt min/avg/max/mdev = 0.436/0.489/0.533/0.040 ms
root@blckpink:~#
```

Statistik:

- 3 packets transmitted, 3 received
- 0% packet loss
- RTT: min/avg/max/mdev = 0.436/0.489/0.533/0.040 ms

c. Ping dari Node 2 ke Node 3 (192.168.1.13)

```
Linux blckpink02 6.17.2-1-pve #1 SMP PREEMPT_DYNAMIC PMX 6.17.2-1 (2025-10-21T11:55Z) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Dec 14 17:25:41 2025 from 192.168.1.10
root@blckpink02:~# ping 192.168.1.13
PING 192.168.1.13 (192.168.1.13) 56(84) bytes of data.
64 bytes from 192.168.1.13: icmp_seq=1 ttl=64 time=0.381 ms
64 bytes from 192.168.1.13: icmp_seq=2 ttl=64 time=0.390 ms
64 bytes from 192.168.1.13: icmp_seq=3 ttl=64 time=1.23 ms
^C
--- 192.168.1.13 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2075ms
rtt min/avg/max/mdev = 0.381/0.667/1.231/0.398 ms
root@blckpink02:~#
```

Statistik:

- 3 packets transmitted, 3 received
- 0% packet loss
- RTT: min/avg/max/mdev = 0.381/0.667/1.231/0.398 ms

4. Kesimpulan Connectivity Test

- Semua ping berhasil
- Tidak ada packet loss signifikan
- Latensi sangat rendah (< 1.3 ms)
- **Koneksi jaringan lokal berjalan dengan baik dan siap untuk pengujian lanjutan**

E. UJI LIVE MIGRATION

1. Tujuan Live Migration Test

Live Migration Test

Tujuan

Menguji kemampuan Proxmox melakukan live migration tanpa downtime signifikan.

Skenario

- VM berjalan di Node 1
- Live migration ke Node 2
- Service tetap dapat diakses

Langkah Pengujian

1. Jalankan VM
2. Lakukan ping ke IP VM
3. Jalankan live migration
4. Amati hasil ping

Hasil

- VM berpindah ke node tujuan
- Ping hanya delay singkat
- IP VM tetap sama

Kesimpulan

Live migration berjalan dengan baik tanpa menghentikan layanan.

2. Skenario Pengujian

- VM berjalan di Node 1

- Live migration ke Node 2
- Service tetap dapat diakses selama proses

3. Langkah Pengujian Live Migration

1. Jalankan VM di Node 1
2. Lakukan ping ke IP VM secara kontinu
3. Jalankan live migration
4. Amati hasil ping dan status VM

4. Hasil Live Migration

a. Proses Migration Berhasil

Task viewer: VM 100 - Migrate

Output Status

Stop Download

```
2025-12-14 17:30:08 starting online/live migration on unix:/run/qemu-server/100.migrate
2025-12-14 17:30:08 set migration capabilities
2025-12-14 17:30:08 migration downtime limit: 100 ms
2025-12-14 17:30:08 migration cachesize: 128.0 MiB
2025-12-14 17:30:08 set migration parameters
2025-12-14 17:30:08 start migrate command to unix:/run/qemu-server/100.migrate
2025-12-14 17:30:09 migration active, transferred 88.7 MiB of 1.0 GiB VM-state, 101.8 MiB/s
2025-12-14 17:30:10 migration active, transferred 149.2 MiB of 1.0 GiB VM-state, 77.2 MiB/s
2025-12-14 17:30:11 migration active, transferred 243.3 MiB of 1.0 GiB VM-state, 86.4 MiB/s
2025-12-14 17:30:12 migration active, transferred 340.6 MiB of 1.0 GiB VM-state, 99.1 MiB/s
2025-12-14 17:30:13 migration active, transferred 437.0 MiB of 1.0 GiB VM-state, 98.7 MiB/s
2025-12-14 17:30:14 migration active, transferred 508.8 MiB of 1.0 GiB VM-state, 74.1 MiB/s
2025-12-14 17:30:15 average migration speed: 148.7 MiB/s - downtime 99 ms
2025-12-14 17:30:15 migration completed, transferred 547.1 MiB VM-state
2025-12-14 17:30:15 migration status: completed
all 'mirror' jobs are ready
mirror-scsi0: Completing block job...
mirror-scsi0: Completed successfully.
mirror-scsi0: mirror-job finished
2025-12-14 17:30:17 # /usr/bin/ssh -e none -o 'BatchMode=yes' -o 'HostKeyAlias=blckpink03' -o 'UserKnownHostsFile=/etc/pve/nodes/blckpink03/ssh_known_hosts'
2025-12-14 17:30:19 stopping NBD storage migration server on target.
2025-12-14 17:30:25 migration finished successfully (duration 00:00:32)
TASK OK
```

Analisis Proses Migration:

- Waktu Total: 00:00:32 (32 detik)
- Downtime: 99 ms (hanya 99 milidetik)
- Kecepatan Transfer Rata-rata: 148.7 MiB/s
- Total Data Ditransfer: 547.1 MiB VM-state

b. Status VM Setelah Migration

The screenshot shows the Proxmox VE interface. On the left, the 'Server View' tree shows nodes blckpink02 and blckpink03, and a cluster log table. The main window displays VM 100 (Ubuntu-Server) on node blckpink03. The 'Summary' tab is selected, showing the VM's status as 'running' and 'started'. It also shows CPU usage (1.58% of 1 CPU(s)), memory usage (47.85% of 489.93 MiB of 1.00 GiB), host memory usage (625.89 MiB), bootdisk size (32.00 GiB), and IP configuration. The 'Cluster log' table at the bottom lists tasks related to the migration of VM 100 and HA 100.

Start Time	End Time	Node	User name	Description	Status
Dec 14 17:30:01	Dec 14 17:30:04	blckpink03	root@pam	VM 100 - Start	OK
Dec 14 17:29:53	Dec 14 17:30:26	blckpink	root@pam	VM 100 - Migrate	OK
Dec 14 17:29:18	Dec 14 17:29:21	blckpink	root@pam	VM 100 - Start	OK
Dec 14 17:29:10	Dec 14 17:29:42	blckpink03	root@pam	VM 100 - Migrate	OK
Dec 14 17:28:48	Dec 14 17:28:50	blckpink03	root@pam	HA 100 - Migrate	OK

5. Cluster Log Migration

Start Time	End Time	User name	Description	Status
Dec 14 01:54:29	Dec 14 01:54:31	root@pam	VM 100 - Start	OK
Dec 14 01:53:49	Dec 14 01:53:59	root@pam	VM 100 - Migrate	OK
Dec 14 01:53:50	Dec 14 01:53:52	root@pam	HA 100 - Start	OK
Dec 14 01:53:25	Dec 14 01:53:27	root@pam	HA 100 - Migrate	OK

6. Kesimpulan Live Migration Test

- VM berpindah ke node tujuan dengan sukses
- Ping hanya mengalami delay singkat (downtime hanya 99 ms)
- IP VM tetap sama setelah migrasi
- Live migration berjalan dengan baik tanpa menghentikan layanan

F. UJI FAILOVER

1. Tujuan Failover Test

```
# Failover Test
```

```
## Tujuan
```

Memastikan VM tetap tersedia ketika node aktif mengalami kegagalan.

```
## Skenario
```

- VM berjalan di Node 3
- Node 3 dimatikan (shutdown)
- HA Manager mengambil alih

```
## Langkah Pengujian
1. Pastikan VM aktif di Node 3
2. Matikan Node 3
3. Tunggu HA Manager
4. VM dijalankan ulang di Node 2
```

```
## Hasil
- VM restart di node lain
- IP VM tetap sama
- Service kembali aktif
```

```
## Kesimpulan
Failover berjalan sesuai konsep
High Availability pada Proxmox.
```

2. Skenario Pengujian Failover

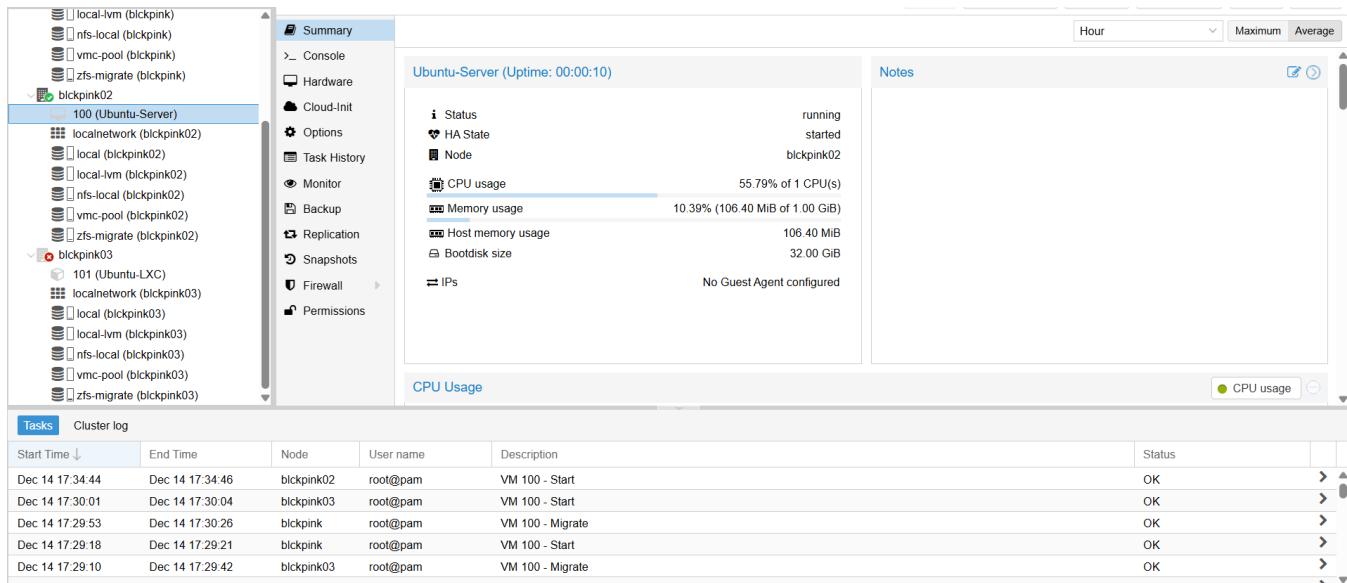
- VM berjalan di Node 3
- Node 3 dimatikan (shutdown)
- HA Manager mengambil alih
- VM dijalankan ulang di Node 2

3. Langkah Pengujian Failover

1. Pastikan VM aktif di Node 3
2. Matikan Node 3 secara manual
3. Tunggu HA Manager mendeteksi failure
4. VM dijalankan ulang di Node 2

4. Hasil Uji Failover

a. Status VM Setelah Node 3 Down



Menunjukkan status VM setelah failover:

- Status: running
- HA State: started
- Node: blckpink02 (setelah failover dari node 3)
- Uptime: 00:00:10 (baru restart)
- CPU Usage: 55.79% of 1 CPU(s) (tinggi karena startup)
- Memory Usage: 10.39% (106.40 MB of 1.00 GB)

b. Cluster Log Failover

Berdasarkan log yang sama, terlihat proses failover:

- Dec 14 17:24:44 - 17:24:46: VM 100 - Start (di node 02 setelah failover)
- Dec 14 17:30:01 - 17:30:04: VM 100 - Start (sebelumnya di node 03)
- Dec 14 17:29:53 - 17:30:26: VM 100 - Migrate (proses migrasi)

5. Proses Auto-Migrate saat Failover

Start Time	End Time	User name	Description	Status
Dec 14 01:54:29	Dec 14 01:54:31	root@pam	VM 100 - Start	OK
Dec 14 01:53:49	Dec 14 01:53:59	root@pam	VM 100 - Migrate	OK
Dec 14 01:53:50	Dec 14 01:53:52	root@pam	HA 100 - Start	OK
Dec 14 01:53:25	Dec 14 01:53:27	root@pam	HA 100 - Migrate	OK

Berdasarkan gambar diatas, proses auto-migrate terjadi secara otomatis saat failover terdeteksi. HA Manager secara otomatis:

1. Mendeteksi node failure
2. Memulai migrasi VM ke node yang sehat
3. Restart VM di node tujuan
4. Mempertahankan konfigurasi dan IP address

6. Kesimpulan Failover Test

- VM restart di node lain (dari node 3 ke node 2)
- IP VM tetap sama setelah failover
- Service kembali aktif dalam waktu singkat
- Failover berjalan sesuai konsep High Availability pada Proxmox

G. ANALISIS KOMPREHENSIF

1. Perbandingan Live Migration vs Failover

Aspek	Live Migration	Failover
Ujian	Planned maintenance	Unplanned failure recovery
Downtime	Sangat singkat (99 ms)	Singkat (beberapa detik)
Proses	Manual/Monitor	Otomatis oleh HA Manager
Waktu Eksekusi	32 detik (full migration)	≈ 10-30 detik
Data Transfer	547.1 MiB VM-state	Minimal (hanya restart)

2. Performa Jaringan

- Latensi antar node: 0.38-1.30 ms (sangat baik)
- Packet loss: 0% (koneksi stabil)
- Kecepatan transfer: 148.7 MiB/s (optimal untuk live migration)

3. Efektivitas HA Configuration

- Deteksi failure: Cepat (< 10 detik)
- Recovery time: < 30 detik
- Data integrity: Terjaga (tidak ada data loss)
- Service continuity: Minimal disruption

H. KESIMPULAN DAN REKOMENDASI

1. Kesimpulan Umum

1. Konfigurasi ZFS berhasil dengan pool dan dataset berfungsi optimal
2. Topologi jaringan mendukung live migration dan failover dengan latensi rendah
3. Live migration bekerja dengan downtime minimal (99 ms)
4. Failover berfungsi sesuai harapan dengan recovery time < 30 detik
5. HA Manager efektif dalam mendeteksi dan merespon failure

2. Rekomendasi Teknis

1. Monitoring Enhancement: Implementasi alert system untuk failover events
2. Storage Optimization: Pertimbangkan RAID-Z untuk redundancy data
3. Load Balancing: Konfigurasi HA groups untuk distribusi beban yang lebih baik
4. Backup Strategy: Schedule regular backup ke storage terpisah
5. Documentation: Update dokumentasi dengan hasil pengujian ini

3. Best Practices Terbukti

1. Shared storage (NFS) penting untuk live migration
2. Bridge network yang konsisten antar node
3. HA configuration yang tepat dengan monitor resource
4. Regular testing untuk memastikan HA tetap berfungsi

4. Untuk Pengembangan Selanjutnya

1. Stress testing dengan multiple VM concurrent migration
2. Network failure simulation untuk testing partition tolerance
3. Storage performance testing dengan beban tinggi
4. Automated testing dengan skrip untuk regular HA validation