

Virtualisasi Server Failover Clustering Menggunakan Network Development Life Cycle Di SMK Negeri 1 Kota Bekasi

Supriyadi ¹, Didik Setiyadi ^{1*}

¹ Teknik Informatika; Universitas Bina Insani; Jl. Siliwangi No.6 Rawa Panjang Bekasi Bekasi Timur 17114 Indonesia, Telp. (021) 824 36 886 / (021) 824 36 996. Fax. (021) 824 009 24; e-mail: daynet.net@gmail.com, didiksetiyadi@binainsani.ac.id

* Korespondensi: e-mail: didiksetiyadi@binainsani.ac.id

Diterima: 10 Desember 2019; Review: 18 Desember 2019; Disetujui: 23 Januari 2020

Cara sitasi: Supriyadi, Setiyadi D. 2019. Implementasi Virtualisasi Server Failover Clustering Menggunakan Network Development Life Cycle di SMK Negeri 1 Kota Bekasi. Jurnal Mahasiswa Bina Insani. 4 (2): 115 – 124.

Abstrak: Teknologi *virtualisasi* dapat membuat sebuah mesin *virtual* dari sesuatu yang bersifat fisik, sehingga dapat menekan biaya pengeluaran mulai dari biaya listrik sampai kepada pengadaan server itu sendiri. Salah satu *virtualisasi* yang berkembang saat ini adalah *Proxmox*. Akan tetapi ada permasalahan yang muncul ketika server *virtualisasi* tersebut mengalami *down* atau *error*. Oleh karenanya sebuah solusi untuk mengatasi permasalahan tersebut dengan menggunakan *High Availability*, yang juga dikenal sebagai *HA Clusters* atau *Failover Clusters*. Dalam perancangan jaringannya dilakukan menggunakan metode *NDLC* yang memiliki tahapan-tahapan dalam penerapan sehingga resiko kegagalan ditahap selanjutnya dapat diminimalisir. Untuk menjamin ketersediaan akses server maka dibutuhkan sebuah tehnik *failover clustering* didalam *virtualisasi server*. Dengan adanya *virtualisasi server* yang menggunakan tehnik *failover clustering* ini maka server tetap dapat diakses oleh siswa SMK Negeri 1 Kota Bekasi untuk menunjang proses Ujian Sekolah Berbasis Komputer.

Kata kunci: *failover, NDLC, proxmox, virtualisasi server*

Abstract: *Virtualization technology able to create a virtual machine from something that have a physical form, so that it can reduce the cost of expenses start from the cost of electricity to the procurement of the server itself. One of the virtualisations that is developing today is Proxmox. However, there are problems that arise when the virtualization server experiences a down or error. Therefor a solution is needed to handle all the problem that mentioned before with a High Availability, which is also known as HA Clusters or in other term is Failover Clusters. At designing the network, it is work with using the NDLC method which has stages in the application so that the risk of failure in the next stage can be minimized. To ensure availability of server access, a failover clustering technique is needed in the server virtualization server. With the server virtualization using this failover clustering technique it can still be accessed by students of SMK Negeri 1 Bekasi City in order to support the Computer-Based School Examination process.*

Keywords: *failover, NDLC, proxmox, virtualisasi server*

1. Pendahuluan

Perkembangan teknologi semakin hari semakin pesat dan itu terjadi pada semua bidang kehidupan. Teknologi tersebut akan mempermudah manusia untuk mencari sesuatu yang diperlukan atau dapat membuat sesuatu menjadi lebih cepat dan mudah. Akan tetapi seiring perkembangannya ditemui masalah-masalah yang muncul seperti halnya kebutuhan untuk penerapan pendidikan yang berbasis Teknologi Informasi (TI). Penerapan pendidikan dalam

bidang TI masih menggunakan pendekatan infrastruktur fisik meliputi satu *hardware* (*processor*, *memory*, *network*, dan *storage*) dalam satu sistem operasi dan satu aplikasi. Penerapan infrastruktur fisik tersebut saat ini telah bergeser dan berubah menjadi sebuah infrastruktur berbentuk *virtual* yang meliputi semua sistem operasi dan multi aplikasi. Saat ini teknologi *virtualisasi* ini telah mengubah arah revolusi industri komputer untuk mengurangi jumlah fisik *hardware*, mengurangi biaya operasional, mengurangi konsumsi fisik, dan perawatan infrastruktur fisik menjadi lebih efisien.

SMK Negeri 1 Kota Bekasi adalah Lembaga pendidikan pemerintah di bidang vokasi, sekolah ini memiliki jurusan TI sejak tahun 2006. Permasalahan yang saat ini dihadapi oleh SMK Negeri 1 Kota Bekasi adalah *server* yang masih menggunakan infrastruktur fisik yang masing-masing *server* diperuntukan untuk satu sistem operasi dan aplikasi *server*. Sehingga jika terjadi kerusakan atau terjadi masalah pada salah satu *server* maka kegiatan akademik seperti USBK (Ujian Sekolah Berbasis Komputer) akan terganggu sampai *server* tersebut dapat berfungsi normal. Fungsi utama yang dimiliki *server* adalah memberikan layanan kepada *client* sehingga *server* dituntut untuk meminimalkan gangguan. Sedangkan *web server* merupakan perangkat lunak yang memberikan layanan berbasis data dalam menerima permintaan *user* melalui *web browser* [1]. Jaringan komputer merupakan sistem yang terdiri dari kumpulan komputer yang didukung oleh perangkat lainnya yang saling terhubung [2].

Penelitian terdahulu mengatakan bahwa Salah satu penyebab kegagalan *server* dalam melayani *client* adalah karena hanya satu *server* sebagai pusat data yang melayani berbagai permintaan data dari komputer *client*. Adapun metode pengembangan sistem dalam penelitian menggunakan metode *Network Development Life Cycle*. Hasil penelitian ini menunjukkan bahwa *server* yang dibangun menggunakan *System Failover Clustering* cukup signifikan dari segi *availability* dan skalabilitas apabila terjadi *failure/downtime* pada *server* yang sedang aktif maka layanan akan berpindah ke *server* pasif kemudian berubah menjadi *server* aktif sehingga akses/permintaan koneksi dari *client* tidak terputus. Kemudian dari segi keamanan data, resiko kehilangan data dapat dihindarkan karena semua data tersimpan pada kedua *server* [3].

Topologi jaringan komputer adalah infrastruktur fisik jaringan komputer yang digunakan untuk mengimplementasikan LAN [4]. *Virtualisasi* adalah sebuah metode yang mengabstraksikan *physical Hardware* dari sebuah Sistem Operasi dan dapat memindahkan Sistem Operasi antara *Physical Hardware* yang berbeda, *virtualisasi* jenis ini disebut dengan *virtualisasi server* [5]. Arti umum dari kata “*availability*” adalah sebuah karakteristik sumber daya baik seseorang atau objek yang dapat diakses atau digunakan. Ketersediaan sumber daya tersebut dapat diukur dengan rasio *time resource* yang dapat diakses atau tidak dapat diakses [6]. Proses *failover* dapat digambarkan seperti sebuah tindakan *switching*, dimana jika terjadi sebuah kegagalan daya listrik atau *network*, dari ketersediaan *resource* untuk melanjutkan operasi dengan *downtime* yang tidak mungkin ada karena tujuan utama *high availability* adalah tidak adanya *downtime* [7].

Kebutuhan informasi sekarang ini membutuhkan layanan yang memiliki teknologi *high availability* untuk menangani gangguan *downtime server*, sehingga dapat menjaga kestabilan *server* dalam melayani permintaan yang tinggi. *Downtime server* dapat terjadi karena adanya kerusakan pada komponen *hardware* maupun *software* yang menyebabkan *crash*, atau perlu dilakukan *upgrade* dan *maintenance* pada *server*. Teknologi yang digunakan untuk menangani *downtime server* adalah *high availability cluster* dengan metode *synchronous replication*, *redundancy*, dan *failover*. Pada pengujian metode *failover*, ketika *master node* mengalami *down*, *slave node* berhasil melakukan *take over* secara otomatis dan menjadi *master node*. Kesimpulan yang diperoleh dari penelitian ini adalah implementasi *high availability cluster* dapat mengurangi waktu *downtime server*, sehingga meminimalkan risiko kehilangan atau kerusakan data [8]. *Failover clustering* menyediakan tingkat kemampuan yang dapat digunakan dalam kasus dimana setiap node dalam cluster dapat mengakses data secara bersamaan. Sebuah faktor kunci untuk merancang *failover clustering*, yaitu tempat penyimpanan yang digunakan oleh node cluster harus dibagi dan dapat diakses oleh setiap *node* yang membutuhkannya [9].

Metodologi yang digunakan adalah *Network Development Life Cycle (NDLC)*. *NDLC* adalah kunci dibalik proses perancangan jaringan komputer, salah satu metode *NDLC* merupakan model mendefinisikan siklus proses pembangunan atau pengembangan sistem jaringan komputer. Kata *cycle* (siklus) adalah kata kunci deskriptif dari siklus hidup

pengembangan sistem jaringan yang menggambarkan secara eksplisit seluruh proses dan tahap pengembangan sistem jaringan yang berkesinambungan [10].

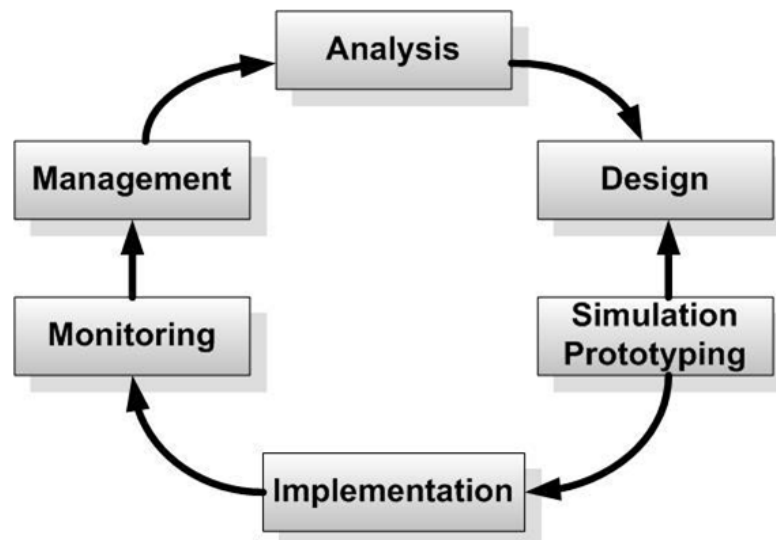
Berdasarkan penelitian-penelitian tersebut penulis mengambil kesimpulan bahwa pada penelitian ini akan dilakukan implementasi *virtualisasi server* menggunakan *proxmox* yang digunakan untuk melakukan penerapan teknik *failover clustering* terhadap *Virtual Machine (VM)* yang didalamnya terdapat aplikasi USBK (Ujian Sekolah Berbasis Komputer) agar terwujudnya *High Availability* sehingga aplikasi tersebut masih bisa diakses ketika *server virtualisasi* mengalami *down* (mati).

2. Metode Penelitian

Model Pengembangan dan kerangka pemikiran dijabarkan pada pembahasan metode penelitian. Pembahasan model pengembangan menjelaskan mengenai model pengembangan yang digunakan untuk melakukan implementasi *virtualisasi server* menggunakan teknik *failover clustering*. Kemudian kerangka pemikiran menjelaskan bagaimana alur penelitian ini dilakukan mulai dari tahap awal sampai penelitian ini selesai dilakukan.

Model Pengembangan

Model Pengembangan dalam penelitian ini adalah dengan model pengembangan Network Development Live Cycle (NDLC). NDLC adalah kunci dibalik proses perancangan jaringan komputer, salah satu metode NDLC merupakan model mendefinisikan siklus proses pembangunan atau pengembangan sistem jaringan komputer [10]. NDLC memiliki tahapan-tahapan, diantaranya proses *analysis*, *design* jaringan, simulasi *prototype*, *implementation*, *monitoring* dan *management* [8].



Sumber : Kurniawan (2017)

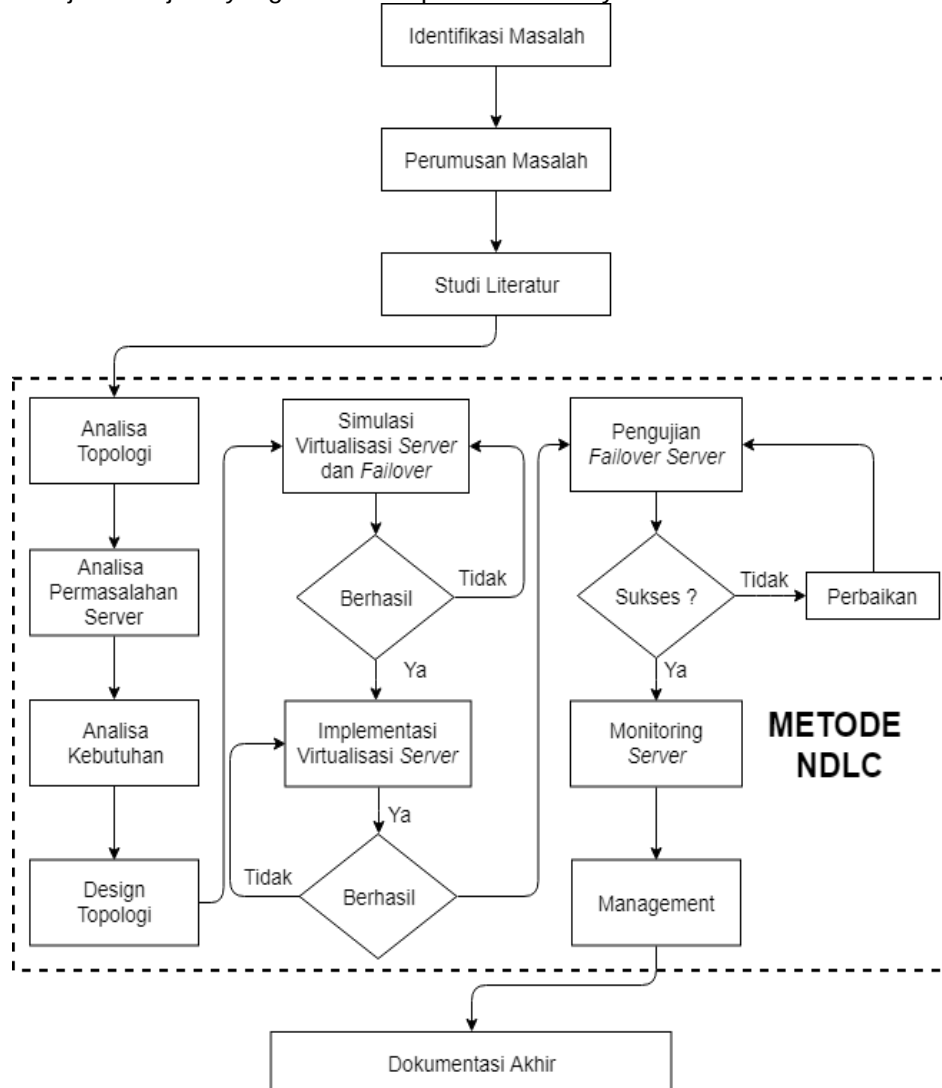
Gambar 1. Metode Network Development Life Cycle (NDLC)

Tahap analisa dilakukan dengan menganalisa *design* topologi berjalan, analisa permasalahan yang muncul dan analisa kebutuhan *hardware* serta *software* yang digunakan untuk penerapan implementasi *virtualisasi server* dengan teknik *failover*. Pada selanjutnya yaitu *design* dengan melakukan pembuatan gambar *design* topologi jaringan *server* yang akan dibangun untuk mendukung pembangunan infrastruktur *virtualisasi server* dan diharapkan dengan gambar ini akan memberikan gambaran seutuhnya dari kebutuhan yang ada. Tahap berikutnya yaitu simulasi *prototype* dalam perancangan infrastruktur jaringan *virtualisasi server* dengan menggunakan bantuan aplikasi *virtualisasi* lainnya seperti VMware yang memiliki kemampuan untuk membuat *virtualisasi* didalam *virtual (nedsted)*. Tahap berikutnya yaitu melakukan proses implementasi dengan proses waktu yang lebih lama dari tahap sebelumnya. Dalam tahap implementasi dilakukan penerapan dari semua yang telah direncanakan dan dirancang sebelumnya. Pada tahap inilah akan terlihat bagaimana jaringan *server* yang sudah dibangun akan memberikan pengaruh terhadap tingkat ketersediaan layanan pada *server* yang telah ada.

Setelah melakukan *implementation*, tahapan selanjutnya adalah *monitoring* yang merupakan tahapan penting agar *virtualisasi server* dengan teknik *failover clustering* dapat berjalan sesuai dengan keinginan dan tujuan awal pada tahap awal analisis, sedangkan pada tahap terakhir dari metode NDLC adalah *Management* atau pengaturan dengan melakukan sebuah kebijakan yaitu dalam hal aktifitas, pemeliharaan dan pengelolaan pada jaringan *server* yang sudah dibangun. Kebijakan perlu dibuat untuk mengatur agar jaringan *server* yang telah dibangun dan berjalan dengan baik dapat berlangsung lama dan unsur *reliability* tetap terjaga.

Kerangka Pemikiran

Penelitian ini dilakukan berdasarkan kerangka pemikiran yang telah dibuat agar penelitian berhasil menjawab tujuan yang telah ditetapkan sebelumnya.



Sumber: Hasil Penelitian (2019)

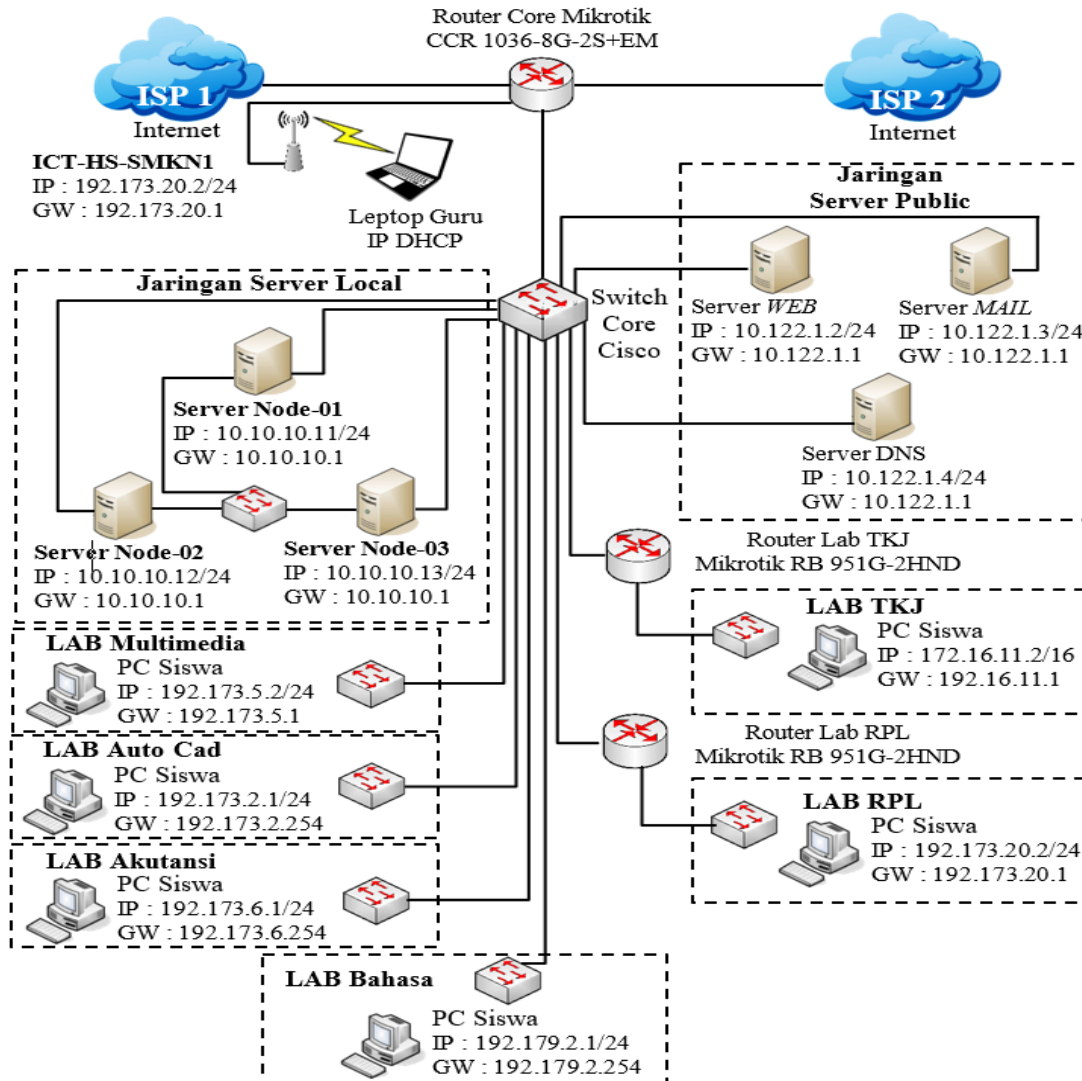
Gambar 2. Kerangka pemikiran penelitian

3. Hasil dan Pembahasan

Pada bagian ini, dijelaskan proses implementasi jaringan *server* sesuai rancangan jaringan *server* yang sudah diusulkan sebelumnya dengan melakukan konfigurasi terhadap *server node*. Implementasi *virtualisasi server* dengan teknik *failover clustering* dilakukan dengan menggunakan perangkat *real hardware* pada *server Dell* dan *web management interface* yang memiliki fungsi untuk meremote *server Proxmox* dengan menggunakan web melalui *operating system windows* pada *client*.

Design Topologi Jaringan Usulan

Pengusulan topologi jaringan yang akan diimplementasikan di SMK Negeri 1 Kota Bekasi tidak merubah bentuk topologi yang sudah ada. Karena perubahan yang terjadi hanya terletak pada bagian *server local* saja. Penjelasan dari topologi yang diusulkan dapat dilihat pada bagian *server local*, terdapat penambahan perangkat yaitu penambahan pada *server* menjadi 3 (tiga) *server*. Penambahan *server* tersebut merupakan *server* cadangan yang digunakan sewaktu-waktu jika *server* yang digunakan mengalami kerusakan. Selain itu juga terdapat perubahan struktur jaringan pada *server* dengan menambah jalur *network* menggunakan *interface* yang tidak terpakai untuk difungsikan sebagai *share storage* (Ceph) guna mendukung implementasi teknik *failover clustering*. Berikut arsitektur jaringan *server* yang diusulkan pada SMK Negeri 1 Kota Bekasi seperti pada gambar 3.



Sumber: Hasil Penelitian (2019)

Gambar 3. Topologi Jaringan Server Usulan

Pada arsitektur jaringan *server* usulan tersebut dapat dilihat bahwa jika *Node-01* mengalami kerusakan atau *down* (mati) maka *storage* dari *Virtual Machine* (VM) yang ada pada *Node-01* secara otomatis akan berpindah ke *Node-02* sehingga *Node-02* akan menjalankan *Virtual Machine* (VM) yang sebelumnya berada di *Node-01*.

Implementasi Virtualisasi Server

Pada proses konfigurasi terhadap rancangan jaringan yang sudah diusulkan sebelumnya. Rancangan jaringan *virtualisasi server* dengan teknik *failover clustering* dilakukan dengan

menggunakan perangkat *real hardware* dengan aplikasi *virtualisasi server Proxmox* dan *web management* yang memiliki fungsi untuk *me-remote* sebuah *server proxmox* melalui *browser* dari *operating system windows*.

Hal yang pertama kali dilakukan adalah mengumpulkan dan memasang seluruh *hardware* yang diperlukan dalam mengimplementasikan *virtualisasi server* dengan teknik *failover clustering*, sesuai dengan rancangan topologi yang diusulkan pada tahap *design*.

Tahap kedua konfigurasi *hosts* pada setiap masing-masing *server node* yang digunakan untuk memudahkan dalam melakukan konfigurasi *cluster* pada *server proxmox*. Untuk melakukan konfigurasi *hosts* pada setiap masing-masing *server node* dilakukan dengan cara *remote* setiap *server* menggunakan *browser*.

Tahap ketiga konfigurasi *proxmox cluster* yang digunakan untuk menjalankan manajemen terpusat dari beberapa *server* fisik dengan menggunakan perintah "*pvecm create*" yang bertindak sebagai *node cluster*. Sedangkan untuk *server node* yang ingin masuk kedalam *cluster* menggunakan perintah "*pvecm add (node cluster)*".

Tahap keempat yaitu melakukan konfigurasi *network* yang digunakan untuk menggabungkan *storage* yang ada pada setiap *server node* dalam sebuah jaringan dengan menggunakan aplikasi *ceph*. Untuk *network storage* ini memiliki jalur yang berbeda dari jalur *remote server node*, hal ini dikarenakan agar tidak membebani jaringan yang digunakan dalam pertukaran *data storage*.

Tahap kelima yaitu melakukan konfigurasi *storage ceph* pada *proxmox* yang digunakan sebagai tempat penyimpanan *virtual drive* ketika membuat *virtual machine* sekaligus sebagai pendukung untuk sistem *failover clustering* di *proxmox*. Sebelum melakukan konfigurasi *storage*, hal yang perlu dilakukan adalah menginstalasi paket *ceph* dengan versi *luminous*.

Tahap keenam yaitu membuat *Kernel-based Virtual Machine (KVM)* pada *proxmox* yang digunakan sebagai tempat *Guest OS* untuk aplikasi ujian sekolah berbasis komputer.

Tahapan terakhir dalam proses implementasi *virtualisasi server* adalah tahapan konfigurasi *failover* yang berfungsi untuk menjamin ketersediaan *server (High Availability)* ketika salah satu *node server* mengalami kerusakan atau *down* (mati).

Pengujian Failover

Pada tahap ini akan dilakukan pengujian *failover* terhadap *Virtual Machine (VM)* dengan melakukan percobaan pada *server node* dengan mematikan salah satu *server node*. Pada percobaan tersebut juga dilakukan perintah *ping client* ke *Virtual Machine (VM)* yang telah dibuat sebelumnya serta *stopwatch* untuk mengetahui berapa lama waktu (*downtime*) yang dibutuhkan agar *client* bisa mengakses *Virtual Machine (VM)* tersebut.

Hasil percobaan tersebut dapat disimpulkan bahwa pengujian metode *failover clustering* pada *virtualisasi server* menggunakan *proxmox* telah berhasil dilakukan yaitu dengan indikasi bahwa *Virtual Machine (VM)* telah berhasil pindah ketika salah satu *node server* mengalami kegagalan atau *down* (mati) maka *node server* dalam satu *cluster* yang sama akan melakukan *take over* (mengambil alih tugas) *Virtual Machine (VM)* tersebut secara otomatis dan menjalankan *Virtual Machine (VM)* seperti yang dilakukan oleh *node server* sebelumnya.

Monitoring

Kegiatan *monitoring* dilakukan untuk mengamati tingkat ketersediaan jaminan layanan *server (availability)* dengan melakukan beberapa analisa pada *server*. Analisa tersebut diambil dari skenario yang telah disusun untuk mengetahui apakah hasil dari implementasi *virtualisasi server* dengan teknik *failover clustering* memberikan hasil yang optimal atau tidak. Kegiatan monitoring ini dilakukan dengan menganalisa tingkat ketersediaan layanan pada *server* khususnya pada *Virtual Machine (VM)* yang terdapat aplikasi USBK.

Analisa *availability* yang dilakukan pada penelitian ini tidak mengambil waktu sebelum terjadi kegagalan sistem ataupun sesudah kegagalan sistem. Tetapi mengamati apakah layanan *server* tetap berjalan saat terjadi kegagalan pada *server virtual* dengan melakukan *monitoring*. Hasil proses *monitoring* tersebut digunakan untuk mengetahui nilai *availability cluster* pada *server* dengan rumus yang digunakan oleh Braastad (2006) dan Calzolari (2009) seperti rumus 1.

$$Availability = \frac{MTBF}{MTBF+MTTR} \times 100\% \dots\dots\dots (1)$$

Mean Time Between Fault (MTBF) adalah nilai rata-rata *uptime* pada server *Virtual Machine* (VM).

Mean Time to Repair (MTTR) adalah nilai rata-rata waktu yang diperlukan untuk mengembalikan layanan pada server *Virtual Machine* (VM). Hasil Perhitungan Availability Cluster tersebut dapat dilihat pada tabel 1.

Tabel 1. Hasil Pengujian Availability

No	Waktu	Penyebab kegagalan	MTBF (Menit)	MTTR (Menit)	Availability
1	17-18 Feb 2019	Node 01 down (Failover) Node 02 down (Failover)	681.35	1.60	99.77%
2	18-19 Feb 2019	Node 01 down (Failover) Node 02 down (Failover) Live Migration	626.29	1.14	99.82%
3	19-20 Feb 2019	Listrik Mati Node 01 down (Failover)	536.82	2.61	99.52%
4	20-21 Feb 2019	Live Migration Node 03 down (Failover)	672.55	0.76	99.89%
5	21-22 Feb 2019	Node 01 down (Failover) Node 02 down (Failover) Node 01 down (Failover)	443.93	1.46	99.67%
Nilai Availability Total			2960.9	7.57	99.75%

Sumber: Hasil Penelitian (2019)

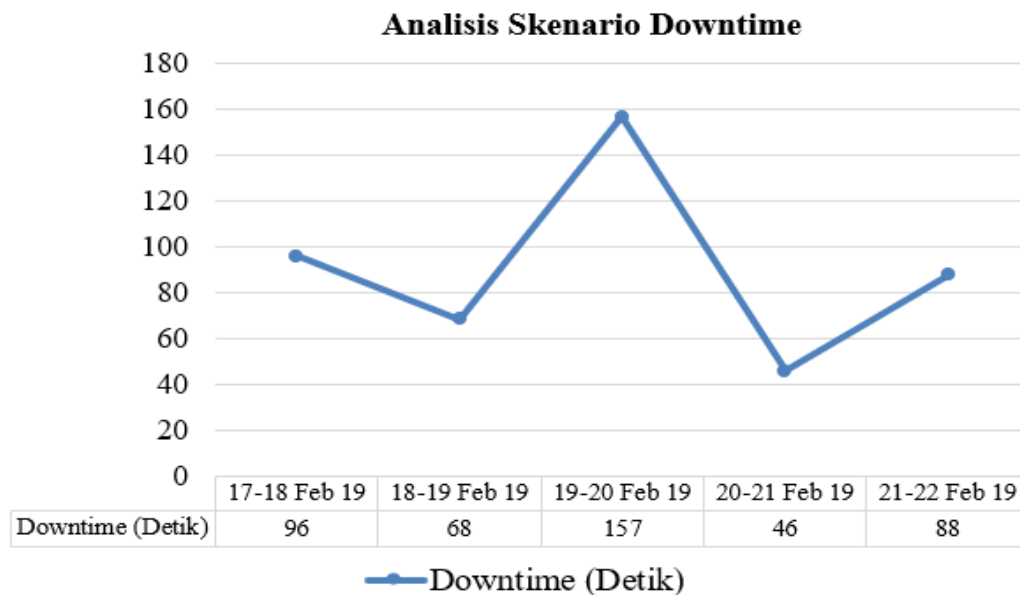
Berdasarkan nilai benchmark pada tabel IV.2 untuk analisa pengujian *availability* hasilnya mendekati 0 dan dapat dikatakan *zero downtime*. Karena *delay (downtime)* yang terjadi pada pengujian *availability* tersebut memiliki nilai rata-rata *delay (downtime)* dalam 5 hari sebesar 1,5 menit dengan tingkat *availability* total mencapai 99.75%.

Analisis Downtime merupakan waktu (*periode of time*) yang terjadi ketika sistem tidak dapat diakses atau digunakan dalam menjalankan fungsinya sesuai yang diharapkan. Analisa pengujian yang dilakukan diambil dari pengujian *availability* dengan menghitung antara waktu sebelum *failover* atau Mean Time Between Faults (MTBF) sampai dengan waktu sebelum mengembalikan layanan atau Mean Time to Repair (MTTR) maka hasilnya dapat dilihat pada tabel 2.

Tabel 2. Hasil Pengujian Downtime

No	Penyebab kegagalan	Sebelum Failover (Menit)	Downtime (Menit)	Failback (Menit)
1	Node 01 down (Failover) Node 02 down (Failover)	681.35	1.60	682.95
2	Node 01 down (Failover) Node 02 down (Failover) Live Migration	626.29	1.14	627.43
3	Listrik Mati Node 01 down (Failover)	536.82	2.61	539.43
4	Live Migration Node 03 down (Failover)	672.55	0.76	673.31
5	Node 01 down (Failover) Node 02 down (Failover) Node 01 down (Failover)	443.93	1.46	445.39

Sumber: Hasil Penelitian (2019)



Sumber: Hasil Penelitian (2019)

Gambar 4. Grafik Hasil Analisa Skenario *Downtime*

Berdasarkan hasil pengujian layanan pada server *virtual machine* (VM) pada tabel 2 dan grafik pada gambar 4 terlihat bahwa besarnya persentase *availability* server mempengaruhi lamanya waktu *downtime* yang dialami *server node*.

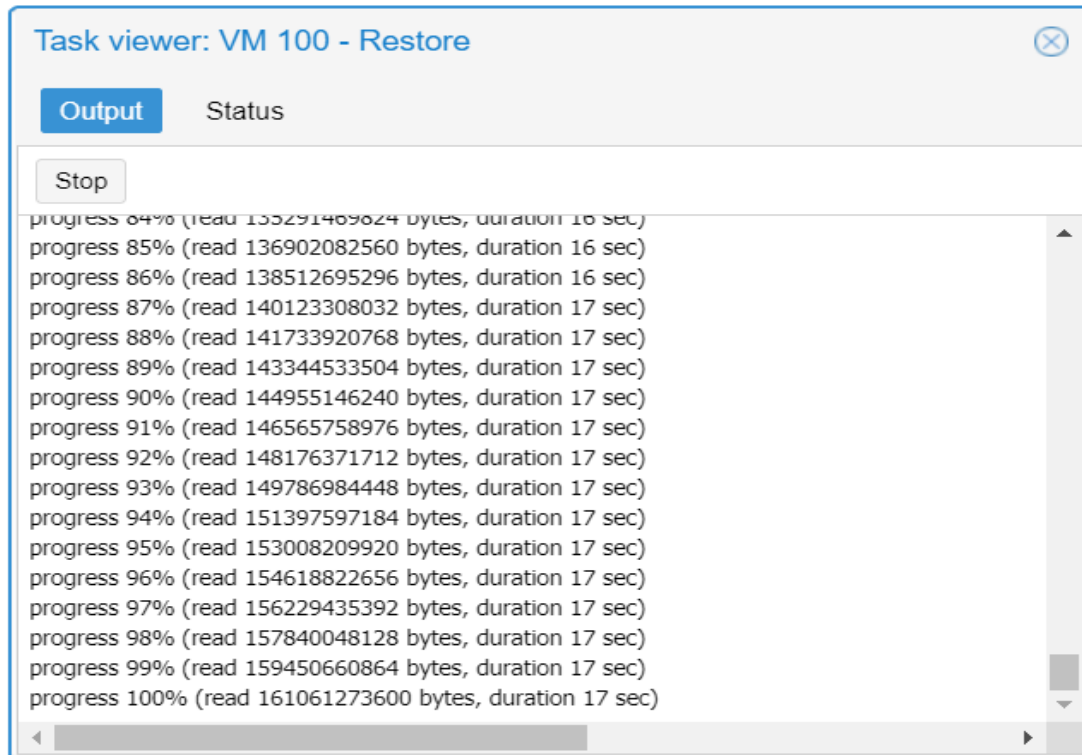
Manajemen

Salah satu yang menjadi perhatian khusus adalah masalah kebijakan (*policy*) yaitu dalam pemeliharaan dan pengelolaan *server*. Kebijakan perlu dibuat untuk mengatur agar sistem yang telah dibangun berjalan dengan baik dan berlangsung lama sehingga *reliability* pada *server* tetap terjaga. Berikut ini ada beberapa pengaturan yang digunakan sebagai pendukung untuk menjaga sistem dapat berjalan dengan baik sehingga lebih optimal yaitu:

Live migration merupakan sebuah fitur dari *proxmox* yang digunakan untuk menghindari *downtime* layanan *server* yang ada di *Virtual Machine* (VM) dengan melakukan pemindahan *Virtual Machine* (VM) tersebut dari *server node-01* ke *server node-02* ketika *server node-01* ingin melakukan *maintenance* perangkat keras baik itu penambahan *hard disk* ataupun *memory*.

Backup dan Restore Virtual Machine (VM)

Untuk mencegah hal-hal yang tidak diinginkan, disarankan untuk melakukan *backup Virtual Machine* (VM) secara berkala. Agar ketika *Virtual Machine* (VM) mengalami kerusakan atau ada konfigurasi yang salah maka bisa menggunakan hasil *backup Virtual Machine* (VM) kedalam *proxmox* tanpa harus membuat ulang *Virtual Machine* (VM) tersebut. *Backup Virtual Machine*, proses untuk melakukan *backup* pada *virtual machine* dilakukan pada saat *virtual machine* dalam keadaan *down* dengan menekan tombol *shutdown* pada *KVM specific menu* milik *Virtual Machine* (VM) 100. langkah selanjutnya adalah mengclick *menu backup* yang ada pada *KVM specific menu* kemudian *click* tombol *Backup now* sehingga akan muncul sebuah *form Backup VM* 100. *Restore Virtual Machine*, pada proses *restore* hal yang harus dilakukan pertama kali adalah menghapus *Virtual Machine* (VM) 100 pada konfigurasi *failover* yang telah dibuat sebelumnya. Setelah itu menghapus *Virtual Machine* (VM) yang bermasalah kemudian membuat kembali *Virtual Machine* (VM) tersebut. Setelah pembuatan *Virtual Machine* (VM) selesai selanjutnya *click Virtual Machine* (VM) yang akan di *restore* kemudian pada *KVM specific menu* *click menu backup* setelah itu *click file* hasil *backup*, Kemudian *click restore* dan akan muncul sebuah *form Restore: VM* 100.



Sumber: Hasil Penelitian (2019)

Gambar 5. Proses Restore pada *Virtual Machine* (VM) 100

4. Kesimpulan

Berdasarkan hasil penelitian serta pengembangan jaringan *server* menggunakan metode *NDLC*, ada beberapa simpulan yang dapat diambil dari implementasi *virtualisasi server* dengan teknik *failover clustering*, yaitu: a). Implementasi *virtualisasi server* dengan teknik *failover clustering* telah berhasil dilakukan dengan menggunakan 3 (tiga) buah *node server* sekaligus pemulihan akses aplikasi USBK yang terinstall pada *Virtual Machine* (VM) ketika salah satu *server node* mengalami *down* pada SMK Negeri 1 Kota Bekasi. b). Penerapan *virtualisasi server* dengan teknik *failover clustering* dapat memberikan dampak pada tingkat ketersediaan *server* sehingga akses terhadap *Virtual Machine* (VM) yang didalamnya terdapat aplikasi USBK akan tetap terjaga apabila salah satu *server node* mengalami *down* karena suatu teknis tertentu pada jaringan *server* di SMK Negeri 1 Kota Bekasi. c). Dengan diterapkannya teknik *live migration* dapat membantu dalam proses perawatan *server* dalam segi penambahan perangkat atau *maintenance server* tanpa harus mematikan *Virtual Machine* (VM) yang didalamnya terdapat aplikasi USBK yang akan diujikan ke siswa di SMK Negeri 1 Kota Bekasi.

Referensi

- [1] Y. Ariyanto, B. Harijanto, and Y. W. Syaifudin, *Virtualisasi Komputer Dalam Pembelajaran Jaringan*. Malang: Polinema Press, 2018.
- [2] W. Komputer, *Cara Mudah Membangun Jaringan Komputer dan Internet*. Jakarta Selatan: MediaKita, 2010.
- [3] Muhajirin, "Optimalisasi Web Server Menggunakan System Failover Clustering Berbasis Cloud Computing," *J. Ilm. Ilmu Komput.*, vol. 3, no. 2, pp. 35–42, 2017, doi: 10.35329/jiik.v3i2.58.
- [4] Kustanto and D. Saputro, *Belajar Jaringan Komputer berbasis Mikrotik OS*. Yogyakarta: Gava Media, 2015.
- [5] W. Panek, *MCSA Windows Server 2016 Study Guide*. United States: Jhon Wiley & Sons, Inc., 2017.
- [6] M. Resman, *CentOS High Availability*. Birmingham: Packt Publising Ltd, 2015.
- [7] G. Canepa, *Centos High Performance*. Birmingham: Packet Publising Ltd, 2016.

- [8] I. Kurniawan, E. Sutanta, and J. Triyono, "Raspberry Pi High Availability Cluster Server dengan Metode Synchronous Replication, Redudancy, dan Failover," *J. JARKOM*, 2017.
- [9] J. Krause, *Mastering Windows Server 2016*. Birmingham: Packet Publising Ltd, 2016.
- [10] R. Goldman, *Learning Proxmox VE*. Birmingham: Packet Publising Ltd, 2016.