

**IMPLEMENTASI *FAILOVER DATABASE*
MENGUNAKAN HEARTBEATS**

SKRIPSI



OLEH

**ANTIKA SARI
NPM. 18020027**

**PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DEHASEN
BENGKULU
2023**

**IMPLEMENTASI *FAILOVER DATABASE*
MENGUNAKAN HEARTBEATS**

SKRIPSI

**ANTIKA SARI
NPM. 18020027**

*Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata I
Pada Program Studi Rekayasa Sistem Komputer Universitas Dehasen Bengkulu*

**PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DEHASEN
BENGKULU
2023**

**IMPLEMENTASI *FAILOVER DATABASE*
MENGUNAKAN HEARTBEATS**

SKRIPSI

OLEH :

**ANTIKA SARI
NPM. 18020027**

DISETUJUI OLEH :

Pembimbing Utama,



**Toibah Umi Kalsum, S.Kom., M.Kom
NIDN.02.060573.01**

Pembimbing Pendamping,



**Riska, S.Kom., M.Kom
NIDN.02.240192.01**

**Mengetahui,
Ketua Program Studi
Rekayasa Sistem Komputer**



**Toibah Umi Kalsum, S.Kom., M.Kom
NIDN.02.060573.01**

**IMPLEMENTASI *FAILOVER DATABASE*
MENGUNAKAN HEARTBEATS**

SKRIPSI





Disusun Oleh :

ANTIKA SARI
NPM. 18020027

Telah Dipertahankan Didepan TIM Penguji
Universitas Dehasen Bengkulu Pada:

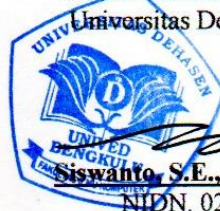
Hari : Sabtu
Tanggal : 17 Juni 2023


Skripsi ini telah diperiksa dan disetujui oleh TIM Penguji:

Penguji	Nama	NIDN	Tanda Tangan
Ketua	Toibah Umi Kalsum, S.Kom., M.Kom	02.060573.01	
Anggota	Riska, S.Kom., M.Kom	02.240192.01	
Anggota	Hendri Alamsyah, S.Kom., M.Kom	02.110391.02	
Anggota	Yessi Mardiana, S.Kom., M.Kom	02.030288.02	

Mengetahui,

Dekan Fakultas Ilmu Komputer
Universitas Dehasen Bengkulu




Siswanto, S.E., S.Kom., M.Kom
NIDN. 02.240363.01

DAFTAR RIWAYAT HIDUP



Penulis bernama Antika Sari, dilahirkan di Desa Arang Sapat Kecamatan Lubuk Sandi Kabupaten Seluma pada tanggal 11 Mei 1999, anak pertama dari tiga bersaudara, Ayah bernama Muhammad dan Ibu bernama Hasni Murni. Menyelesaikan pendidikan di Sekolah Dasar Negeri (SDN) 144 Desa Arang Sapat pada tahun 2011, kemudian penulis melanjutkan pendidikan pada SMP BP Pancasila Bengkulu selesai pada tahun 2014 dan selanjutnya menyelesaikan pendidikan SMK Negeri 3 Tahun 2017. Kemudian Penulis melanjutkan pendidikan ke perguruan tinggi yaitu pada Universitas Dehasen (UNIVED) Bengkulu dengan mengambil Jurusan Rekayasa Sistem Komputer pada Fakultas Ilmu Komputer, untuk jenjang Strata Satu (S-1) pada tahun 2018.

MOTTO DAN PERSEMBAHAN

MOTTO

- ❖ Sesungguhnya bersama kesukaran itu ada keringanan, karena itu bila kau sudah selesai (mengerjakan yang lain), dan berharaplah kepada tuhanmu (Q.S Al Insyariah : 6-8)
- ❖ Yakin, Ikhlas, Istiqomah, dan Bersyukur
 - Berangkat dengan penuh keyakinan
 - Berjalan dengan penuh keikhlasan
 - Istiqomah dalam menghadapi cobaan
 - Dan bersyukur atas apa yang telah diperoleh (Penulis)

PERSEMBAHAN

Skripsi ini kupersembahkan kepada:

- ❖ Suami tercinta (Gige Ginanjar Putra) yang selalu setia menemani dan selalu memberikan supoort agar aku mengejar cita-cita
- ❖ Orang Tua Tercinta (Muhammad dan Hasni Murni) Sebagai tanda bakti, hormat dan rasa terimakasih yang tiada terhingga kupersembahkan karya kecil ini kepada Ayah Ibu yang telah memberikan kasih sayang yang amat sangat tulus untukku, dukungan yang tiada henti, doa yang selalu dipanjatkan untuk kebaikan dan kebahagiaanku, Ayah Ibu adalah Inspirasiku, Motivasiiku, guru terbaikku dan semangatku. Meskipun ayah dan ibu telah tiada aku tetap barharap bisa mencapai semua keinginan Ayah dan Ibu yang tertunda, doakan anakmu dari dunia sana.....
- ❖ Saudaraku tercinta, meskipun hanya kalian yang aku punya pada saat ini, terima kasih telah menjadi pengganti orang tua bagi ku.
- ❖ Almamaterku.

SURAT PERNYATAAN ORSINILITAS & PERSETUJUAN PUBLIKASI
AKADEMI SKRIPSI

Yang bertanda tangan di bawah ini:

Nama : Antika Sari
NPM : 18020027
Program Studi : Rekayasa Sistem Komputer
Fakultas : Ilmu Komputer
Tempat/Tgl Lahir : Arang Sapat/ 11 Mei 1999
Alamat : Jl. Semangka, Padang Serai - Bengkulu

Dengan ini menyatakan dengan sesungguhnya bahwa SKRIPSI dengan judul :

**IMPLEMENTASI FAILOVER DATA BASE MENGGUNAKAN
HEARTBEATS**

1. Adalah benar dibuat oleh saya sendiri untuk memenuhi persyaratan kelulusan akademi.
2. Pada bagian-bagian tertentu dalam penulisan skripsi yang saya kutip dari hasil karya orang lain telah ditulis sumbernya secara jelas sesuai dengan norma, kaidah dan etika penulisan ilmiah.
3. Jika dikemudian hari diketahui bukti berdasarkan bukti-bukti yang kuat ternyata skripsi tersebut dibuat oleh orang lain atau diketahui bahwa skripsi tersebut merupakan plagiat/mencontek/menjiplak hasil karya ilmiah orang lain, maka dengan ini saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi-sanksi lainnya sesuai dengan peraturan yang berlaku.
4. Dan atas pernyataan orisinilitas tersebut di atas, maka saya menyetujui untuk memberi kepada Universitas Dehasen Bengkulu atas bebas royalti non eksklusif untuk menyimpan, mengalih mediakan. Mendistribusikan dan mempublikasikan skripsi saya tanpa perlu meminta izin, selama mencantumkan nama saya sebagai penulis.
5. Saya bersedia menanggung secara pribadi tanpa melibatkan pihak Universitas Dehasen Bengkulu segala bentuk tuntutan hukum yang timbul atas pelanggaran hak cipta dalam karya ilmiah saya ini.

Demikian surat pernyataan ini dibuat untuk dipergunakan sebagaimana mestinya.

Bengkulu, 12 Juni 2023



(Antika Sari)
NPM. 18020027

ABSTRAK
IMPLEMENTASI FAILOVER DATABASE
MENGGUNAKAN HEARTBEATS

Oleh:

Antika Sari¹

Toibah Umi Kalsum, M.Kom²

Riska, S.Kom., M.Kom²

Untuk menerapkan *high availability server* ini dapat menggunakan *software* aplikasi khusus untuk *clustering* seperti *heartbeat*. *Heartbeat* ini Merupakan bagian dari proyek *linux high availability*. Dengan menggunakan *heartbeat* maka dapat melakukan *failover* dengan mempromosikan *server* aktif untuk melayani *request* dari *client* dan menggantikan fungsi dari *server* utama. Hal ini dilakukan untuk menjaga agar ketersediaan data yang ada pada *server data base* selalu tersedia saat ada *request* oleh *client*. Tujuan penelitian ini untuk membangun *Server database* yang selalu tersedia dengan menerapkan *failover server* dan menerapkan sistem replikasi data pada *server database*. Dengan teknologi *failover database* menggunakan *heartbeats* dengan menggunakan sistem replikasi master to master maka ketersediaan data dan kinerja sistem meningkat karena data telah memiliki cadangan. *Failover database* menggunakan *heartbeats* dengan menggunakan sistem replikasi master to master memiliki kelebihan (dapat melakukan backup data secara otomatis) serta kekurangan (membutuhkan perangkat tambahan), sehingga untuk menggunakan teknologi ini perlu disesuaikan dengan kebutuhan pengembangan sistem

Kata Kunci: Failover Database, Heartbeats

Keterangan :

1: Peneliti

2: Pembimbing 1 dan Pembimbing 2

ABSTRACT

THE IMPLEMENTATION OF FAILOVER DATABASE USING HEARTBEATS

By:
*Anika Sari*¹
*Toibah Umi Kalsum*²
*Riska*²

For implement this high availability server, you can use special application software for clustering, such as heartbeat. This heartbeat is part of linux high availability project. Using heartbeat, you can do failover by promoting an active server to serve requests from clients and replace the function of the main server. This is done to ensure that the availability of existing data on the database server is always available when there is a request by the client. The purpose of this research is to build a database server that is always available by implementing a failover server and implementing data replication system on the database server. Database failover technology using heartbeats and using master to master replication system, data availability and system performance increase because the data has a backup. Failover database using heartbeats using master to master replication system has advantages (can do data backup automatically) and disadvantages (requires additional devices), so using this technology needs to be adjusted to the needs of system development

Keywords: Failover Data Base, Heartbeats.

Information:

- 1: Student
- 2: Supervisor I and Co-Supervisor



KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan kehadiran Allah SWT, karena berkat ridho, rahmat, dan hidayah-Nya maka skripsi yang berjudul “Implementasi failover Database Menggunakan Heartbeats” dapat diselesaikan sebagaimana mestinya..

Dalam menyelesaikan skripsi ini, penulis menyadari masih banyak mengalami kesulitan, namun atas kerja keras dan bimbingan, petunjuk serta pengarahan dari berbagai pihak akhirnya penulis dapat menyelesaikan penelitian ini.

Pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada berbagai pihak yang telah banyak membantu, memberikan bimbingan, dan semangat serta do'a sehingga penelitian ini dapat diselesaikan, oleh sebab itu izinkanlah penulis menyampaikan rasa terima kasih kepada :

1. Prof. Dr. Husaini, S.E., M.M., Ak, CA, CRP selaku Rektor Universitas Dehasen (UNIVED) Bengkulu.
2. Bapak Siswanto, S.E., S.Kom., M.Kom selaku Dekan Fakultas Ilmu Komputer di Universitas Dehasen (UNIVED) Bengkulu.
3. Ibu Toibah Umi Kalsum, S.Kom, M.Kom selaku Ketua Program Studi Rekayasa Sistem Komputer di Universitas Dehasen (UNIVED) serta selaku Dosen pembimbing satu dalam skripsi ini.
4. Bapak Riska, S.Kom., M.Kom Selaku Dosen Pembimbing II (dua) dalam skripsi ini.

5. Kedua orang tuaku yang memberikan dorongan serta semangat dan do'a kepadaku.
6. Rekan-rekan dan semua pihak yang turut memberikan saran dan kritikan sehingga skripsi ini dapat diselesaikan dengan baik.

Penulis menyadari bahwa dalam penulisan proposal skripsi ini jauh dari sempurna, sehingga banyak kekurangan dan kesalahan, maka dari itu penulis sangat memerlukan kritik dan saran yang sifatnya membangun dari berbagai pihak untuk kesempurnaan proposal skripsi ini.

Akhirnya penulis berharap agar skripsi ini dapat berguna dan bermanfaat bagi mahasiswa Universitas Dehasen Bengkulu khususnya dan pembaca pada umumnya.

Bengkulu, September 2023

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
LEMBAR PERSETUJUAN	iv
DAFTAR RIWAYAT HIDUP	v
MOTTO	vi
PERSEMBAHAN.....	vii
ABSTRAK	viii
ABSTRACT	ix
KATA PENGANTAR.....	x
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
BAB II LANDASAN TEORI	
2.1 Implementasi	5
2.2 <i>High Availability</i>	5
2.3 <i>Failover</i>	7
2.4 Heartbeat	8
2.5 Database	9
2.6 Web Server	9
2.7 HAProxy.....	10
2.8 Jaringan Komputer	11
2.9 Ubuntu Server	20

BAB III METODOLOGI PENELITIAN

3.1 Subjek Penelitian	22
3.2 Metode Penelitian	23
3.3 Instrumen Perangkat Lunak dan Perangkat Keras	25
3.4 Metode Pengumpulan Data.....	26
3.5 Metode Perancangan Sistem.....	27
3.6 Rancangan Pengujian.....	31

BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian.....	33
4.1.1 Pengujian Ketersediaan (<i>Availability</i>).....	35
4.1.2 Pengujian <i>Downtime</i> dan <i>Resptime</i>	39
4.1.3 Pengujian Waktu Sinkronisasi Database.....	41
4.2 Pembahasan.....	45
4.2.1 Instalasi Linux Ubuntu Server.....	46
4.2.2 Instalasi LAMP (Apache2, MySQL dan PHP)	50
4.2.3 Install Heartbeats.....	52
4.2.4 Arsitektur Replikasi.....	54
4.2.5 Konfigurasi Replikasi.....	54
4.2.6 Membuat Database	54
4.2.7 Mengkonfigurasi My.ini.....	55
4.2.8 Membuat User Replikasi	56
4.2.9 Meng ubungkan Master 1 Dengan Master 2 Agar Dapat berkomunikasi.....	57
4.3 Hasil Pengujian.....	58

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan	67
5.2 Saran	67

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar	Halaman
2.1 Konsep Heartbeat.....	8
2.2 Topologi Bus	14
2.3 Topologi Ring	15
2.4 Topologi Star	15
2.5 Topologi Mesh.....	16
3.1 NDLC Model.....	23
3.2 Diagram Blok Sistem Baru	28
3.4 Rencana Kerja Sistem	29
4.1 Tampilan Status Server.....	34
4.2 Tampilan Ping ke Server Aplikasi 1	34
4.3 Tampilan Ping ke Server Aplikasi 2.....	35
4.4 Tampilan Server A Mati Dari Terminal Linux.....	35
4.5 Tampilan Status Server.....	36
4.6 Tampilan Ping ke Server Aplikasi 1	36
4.7 Tampilan Ping ke Server Aplikasi 2.....	36
4.8 Tampilan Status Server.....	37
4.9 Tampilan Ping ke Server Aplikasi 1	37
4.10 Tampilan Server A Mati Dari Terminal Linux.....	38
4.11 Tampilan Informasi Server 1	40
4.12 Tampilan Informasi Server 2	40
4.13 Tampilan Informasi Status Server 1	41
4.14 Tampilan Informasi Status Server 2	42
4.15 Tampilan Status Question Server 1	42
4.16 Tampilan Status Question Server 2	43
4.17 Tampilan Data Pada Aplikasi Server 1	43
4.18 Tampilan Data Pada Database Server 1.....	44
4.19 Tampilan Informasi Server A	45
4.20 Tampilan Informasi Server B	45

4.21	Tampilan Piihan Bahasa Install Linux.....	46
4.22	Tampilan Input Proxy Linux	47
4.23	Tampilan Pilihan Resipotory Linux	47
4.24	Tampilan Pilihan HDD Sistem Linux	48
4.25	Tampilan Konfirmasi Pilihan HDD Sistem Linux	48
4.26	Tampilan Proses Install Linux	49
4.27	Tampilan Install Linux Selesai	49
4.28	Tampilan Status Apache2.....	50
4.29	Tampilan Proses Install MySQL	51
4.30	Tampilan Intallasi PHP.....	51
4.31	Tampilan Konfigurasi File ha.cf Heartbeat	52
4.32	Tampilan Konfigurasi File Authkey.....	53
4.33	Tampilan Server Heartbeat.....	58
4.34	Tampilan Server 1	59
4.35	Tampilan Server 2	60
4.36	Tampilan Status Server.....	63
4.37	Tampilan Informasi Server 1	64
4.38	Tampilan Informasi Server 2	64
4.39	Tampilan Aplikasi Yang di Gunakan	65

DAFTAR TABEL

Tabel	Halaman
2.1 Lapisan Model OSI.....	18
3.1 Spesifikasi Perangkat Keras	26
3.2 Pengujian dan Analisa	32
4.1 Skenario Pengujian	33
4.2 Hasil Perhitungan Availability	38
4.3 Waktu Delay Pindahan Node.....	39
4.4 Perhitungan Availability	61
4.5 Hasil Perhitungan Availability	61
4.6 Waktu Delay Pindahan Node.....	62
4.7 Hasil Pengujian.....	66

DAFTAR LAMPIRAN

Lampiran

1. Time Schedule
2. Struktur Organisasi

BAB I PENDAHULUAN

1.6 Latar Belakang

Pada era perkembangan teknologi informasi yang sangat pesat, ketersediaan akses data dan informasi merupakan bagian terpenting saat ini. Untuk menunjang ketersediaan akses data dan informasi dibutuhkan koneksi internet yang selalu aktif dan juga server dari pengelola data dan informasi yang selalu online setiap saat. Namun banyak faktor yang dapat menyebabkan server ini tidak tersedia setiap saat, seperti kegagalan fungsi Hardware dan software .

Untuk mengatasi hal itu, ada cara yang dapat diterapkan yaitu menggunakan server dengan teknik ketersediaan tinggi atau lebih dikenal dengan sebutan *high availability server*. Implementasi *high availability server* dapat dilakukan dengan cara memasang perangkat lunak kluster pada *server master* dan *server backup*. *High availability server* dapat terus berjalan untuk melayani *request* dari pengguna karena menerapkan teknik *failover*. Teknik *failover* diterapkan pada *server* duplikasi dan *server* duplikasi tersebut dijadikan satu

kluster dengan *server* utama. Tujuan *failover* adalah agar pengguna tetap dapat mengakses ke *server* walaupun *server* sedang mengalami kegagalan *software*, *hardware* atau lainnya yang mengakibatkan *server* tersebut tidak dapat diakses.

Untuk menerapkan *high availability server* ini dapat menggunakan *software* aplikasi khusus untuk *clustering* seperti heartbeat. Heartbeat ini

Merupakan bagian dari proyek linux *high availability*. Dengan menggunakan heartbeat maka dapat melakukan *failover* dengan mempromosikan *server* aktif untuk melayani *request* dari *client* dan menggantikan fungsi dari *server* utama. Hal ini dilakukan untuk menjaga agar ketersediaan data yang ada pada *server data base* selalu tersedia saat ada *request* oleh *client*. Secara sederhana, cara kerja *failover* adalah jika *server master* mengalami kegagalan maka secara otomatis *server backup* akan mengambil alih tugas *server master*. Dengan kata lain implementasi *failover* ini dapat menghindari *single point of failure*, dimana *single point of failure* adalah sebuah komponen penting dalam sistem yang dimana jika komponen ini berhenti maka keseluruhan sistem akan berhenti pula.

Dari uraian latar belakang di atas, penulis tertarik untuk mengangkat judul penelitian “*Implementasi failover Data Base Menggunakan Heartbeats*”.

1.7 Rumusan Masalah

Dari latar belakang diatas, maka rumusan masalah dalam penelitian ini adalah bagaimana mengimplementasikan *failover data base* menggunakan Heartbeats?

1.8 Batasan Masalah

Dari latar belakang dan rumusan masalah yang sudah penulis jabarkan diatas, batasan masalah dalam penelitian ini adalah sebagai berikut.

1. *Failover* database akan dijalankan pada jaringan intranet (Lokal).
2. Server menggunakan sistem operasi linux ubuntu *server*

3. Arsitektur jaringan menggunakan:
 - a. 1 *Server* Utama
 - b. 1 *Server* Cadangan
 - c. 1 *Load balancer* (*Server* Aplikasi)
4. Fokus pengujian terdiri dari:
 - a. Pengujian *Availability*
 - b. Pengujian *Downtime* dan *Resptime*
 - c. Pengujian Waktu Sinkronisasi data base

1.9 Tujuan Penelitian

A. Tujuan Umum

Untuk memenuhi syarat melanjutkan skripsi pada Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer Universitas Dehasen Bengkulu.

B. Tujuan Khusus

2. Membangun *Server database* yang selalu tersedia dengan menerapkan *failover server*.
3. Menerapkan sistem replikasi data pada *server database*.

1.10 Manfaat Penelitian

A. Bagi Penulis

Dapat dijadikan sebagai bahan pembelajaran dalam perancangan *database server* yang selalu tersedia dalam proses melayani *request* dari *client*.

B. Bagi Pembaca

Dapat dijadikan sebagai referensi untuk penulisan berikutnya yang berhubungan dengan jaringan komputer terutama perancangan database *server* dengan penerapan *failover* menggunakan heartbeat pada jaringan komputer.

BAB II

LANDASAN TEORI

2.1 Implementasi

Menurut Amin (2017:117), implementasi adalah tahapan penerapan atau tindakan yang diperlukan agar mencapai sukses dalam suatu penelitian. Oleh karenanya, tahapan ini bukan lagi sebagai wacana pemikiran atau ide lagi, tetapi sudah berada pada tahapan perilaku dan tindakan yang diperlukan dalam penelitian.

Menurut Setiawan (2015:78), implementasi merupakan penerapan aplikasi dari hasil perancangan sistem yang ada untuk mencapai suatu tujuan yang diinginkan. Implementasi melaksanakan perintah-perintah yang secara terstruktur dari awal sampai akhir.

Dari kedua pengertian di atas, dapat disimpulkan bahwa implementasi merupakan penerapan dari perancangan yang diperlukan agar mencapai sukses dalam suatu penelitian.

2.2 *High Availability*

Menurut Pratama (2021:5573), *High availability* dalam teknologi informasi dapat diartikan sebagai sebuah sistem atau komponen yang dapat beroperasi secara terus menerus dalam jangka waktu yang lama tanpa mengalami gangguan. Pada dasarnya, *high availability* mengimplementasikan satu atau lebih *server backup* dalam mode *standby*, yang kemudian dapat menjadi siaga dalam beberapa saat ketika ditemukan kegagalan dalam *server master*.

Menurut Purnomo dan Edi (2017:2), *High Availability* merupakan *server* berbeda berkerja bersama-sama untuk memastikan *downtime* pada *resource* dikurangi seminimal mungkin. Tujuan dari *High Availability Cluster* adalah untuk memastikan bahwa sumber daya mencapai *Availability* semaksimal mungkin, Jika *server down* atau jika sumber daya yang berhenti, HA *cluster* akan memonitor dan memastikan *resource* atau sumber daya dihidupkan ulang pada tempat lain dalam sistem *cluster*, sehingga dapat digunakan lagi setelah mengalami gangguan. Konsep tersebut berkaitan dengan kemampuan sistem untuk mengatasi terjadinya gangguan, kerusakan *hardware*, *crash/down*, kesalahan jaringan bahkan kegagalan *server* yang di sebabkan *software* yang gagal melakukan tugas semestinya. Solusi yang ditawarkan berupa backup data atau *failover* data yang dilakukan secara *real time*. Saat *server* utama berhenti berjalan, maka *server slave* akan mengambil alih peran *server* utama dengan kualitas penanganan *input* dan *output* yang sama dengan *server* utama. Sistem akan selalu melakukan sinkronisasi data diantara keduanya atau mungkin lebih untuk mendapatkan *redundancy* data.

Pengujian *availability* atau ketersediaan tinggi terhadap *server* dapat dilakukan dengan menggunakan rumus berikut. Berikut.

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

Dimana,

- a. *Mean Time Between Faults* (MTBF) merupakan waktu rata-rata *uptime*.
- b. *Mean Time to Repair* (MTTR) merupakan rata-rata waktu yang diperlukan untuk mengembalikan layanan setelah mengalami *serverdown* sampai *server backup* menangani dan memberikan layanan kembali.

2.3 *Failover*

Menurut Pratama (2021:5573), *Failover* merupakan sebuah teknik untuk menghindari kegagalan pada sistem. Dengan implementasi *failover* pada sistem, sistem akan terus berjalan untuk melayani pengguna karena *failover* memindahkan *traffic* dari komponen yang mengalami kegagalan ke komponen *backup*. Proses *failover* dapat dirancang secepat mungkin setelah saat terjadi kegagalan. Didalam *failover* terdapat proses *failback*, *failback* adalah proses kembalinya komponen yang sebelumnya mengalami kegagalan. Terdapat dua model dari *failover*, yaitu *failover active-active* dan *failover active-standby*.

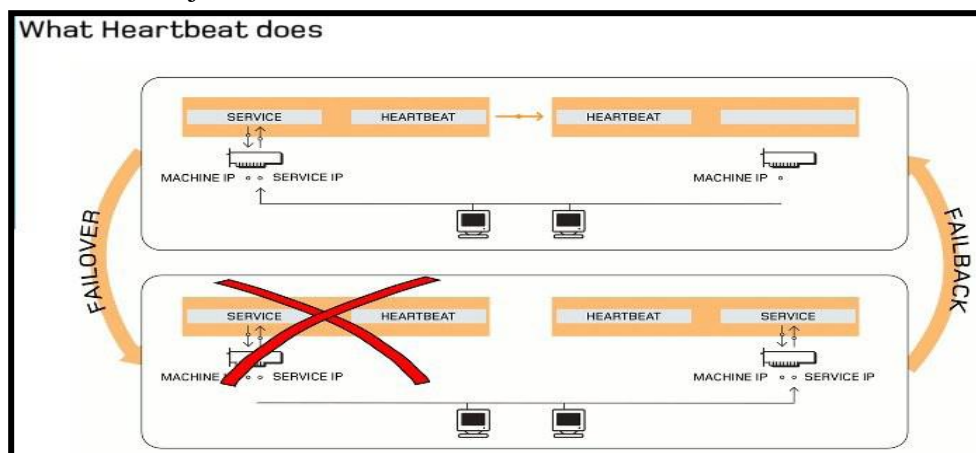
Menurut Purnomo dan Edi (2017:2), *Failover* merupakan sistem komunikasi dua atau lebih *server* ditempat yang berbeda yang dapat saling *backup* data dan mampu menggantikan pelayanan bila *server* lain down. *Failover* bertujuan untuk membantu menjaga akses client ke sumber daya di *server*, ketika *server* mengalami kegagalan fungsi *software*, atau kegagalan akses *server*. *Failover cluster* merupakan sekumpulan *server* yang saling bekerjasama untuk memberikan pelayanan meskipun berada ditempat yang berbeda, dan memiliki kualitas data atau sumberdaya yang sama antara *server* yang satu dengan *server* lainnya. Sistem *Failover* akan bekerja untuk menghubungi *server-server* yang menjadi anggota *cluster*-nya untuk mengambil alih tugas *server* utama saat terjadi kegagalan fungsi dalam waktu tertentu

2.4 Heartbeat

Menurut Pratama (2021:5574), Heartbeat merupakan sebuah perangkat lunak *failover* berbasis klaster, yang memastikan bahwa *resources* akan selalu tersedia untuk diakses. Dalam jangka waktu tertentu, masing-masing *node* dari Heartbeat akan mengirimkan packet melewati jaringan kepada *node* Heartbeat lain sebagai sinyal tetap hidup. Jika tidak ada pesan Heartbeat dari *node master*, maka diasumsikan *node master* telah mati kemudian *node backup* menjalankan teknik *failover* serta mengambil alih peran dari *node master*.

Menurut Afif dan Suryono (2013:66), Heartbeat merupakan aplikasi dasar untuk Linux-HA (*Linux High Availability*). Heartbeat akan menjalankan script inisialisasi untuk HA dan saat *node* atau *server* mati dan hidup. *Heartbeat* juga melakukan perpindahan IP dari satu *node* ke *node* yang lain (*IP floating*).

Komunikasi heartbeat bisa dilakukan melalui serial ports, UDP/IP *broadcast* (*ethernet*, etc), UDP/IP *multicast* (*ethernet*). Selain itu heartbeat juga menangani *service-service* apa saja yang akan dijalankan pada saat *node* atau *server* menjadi aktif.



Gambar 2.1 Konsep Heartbeat

2.5 Database Server

Menurut Rahmatulloh dan MSN (2017:243), *Database server* adalah aplikasi komputer yang menyediakan layanan data ke komputer atau program komputer dan memiliki fungsi sebagai tempat penyimpanan data, seperti yang ditetapkan oleh model klien-server. Istilah ini juga merujuk kepada sebuah komputer yang didedikasikan untuk menjalankan program *server database*. Salah satu contoh aplikasi database adalah *MySQL*, Oracle, MariaDB dan lain sebagainya.

Menurut Darmawan, dkk (2018:94), MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Structured Query Language) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

2.6 Web Server

Menurut Rahmatulloh dan MSN (2017:242), *Web server* adalah sebuah *software* yang memberikan layanan berbasis data dengan menggunakan protokol HTTP atau HTTPS dari *client* menggunakan aplikasi web *browser* untuk *request* data dan *server* akan mengirim data dalam bentuk halaman web dan pada umumnya berbentuk dokumen HTML. Halaman web yang diminta bisa terdiri dari berkas teks, video, gambar, file dan banyak lagi. Salah satu program dari *Web Server* adalah Apache. Apache merupakan web server yang paling banyak dipergunakan di Internet. Program

ini pertama kali didesain untuk sistem operasi lingkungan UNIX, untuk saat ini telah tersedia apache yang di desai untuk sistem operasi lainnya. Apache mempunyai program pendukung yang cukup banyak. Hal ini memberikan layanan yang cukup lengkap bagi penggunaanya.

Menurut Masykur dan Prasetyowati (2016:95), Apache adalah sebuah nama web *server* yang bertanggung jawab pada permintaan dan jawaban HTTP dan *logging* informasi secara lebih detail. Selain itu, Apache juga merupakan suatu web *server* yang *modular*, mengikuti standar protokol HTTP dan sangat disukai. Apache adalah sebuah web *server* yang dapat dijalankan pada banyak sistem operasi seperti, Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta Platform lainnya. Dan digunakan untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP.

2.7 HAProxy

Menurut Noviyanto, dkk (2015:24), HAproxy adalah produk *open source* yang mendukung keperluan penyeimbang beban dan *failover* web *server*. HAProxy ini banyak digunakan untuk keperluan *reverse proxy* di site-site yang trafik hariannya tinggi. HAProxy sangat diperlukan apabila aplikasi menuntut SLA yang cukup ketat dan tidak mentolerir adanya *downtime*. Selain itu HAProxy memungkinkan adanya *backup* dari *load balancer* yang digunakan sehingga apabila terjadi *down* pada *load balancer* utama maka *load balancer* yang menjadi *backup* dalam waktu singkat secara otomatis akan menggantikan *load balancer* yang mati tersebut. Pada HAProxy ini memiliki beberapa parameter di dalamnya yaitu Global parameter, Proxy,

Konfigurasi *Server*, Manipulasi HTTP, *Accesslist*, *Logging*, Statistik dan *monitoring*.

2.8 Jaringan Komputer

Menurut Madcoms dalam Rahadjeng dan Ritapuspitari (2018:53), Jaringan komputer merupakan kumpulan dari beberapa komputer dan peralatan penunjang lainnya yang terhubung dalam satu kesatuan dan saling terkoneksi.

Menurut Foruzen dalam Pratama (2014:21), Jaringan komputer merupakan hasil dari koneksi (hubungan) dari sejumlah perangkat atau komputer yang dapat saling berkomunikasi satu sama lain. Perangkat yang dimaksud pada definisi ini mencakup semua jenis perangkat komputer (komputer desktop, komputer jinjing, smartphone, PC tablet) dan perangkat penghubung (*router, switch, modem, wireless access point*).

Pada sebuah jaringan komputer minimal terdapat dua buah komputer atau perangkat yang saling terhubung satu sama lain. Dalam sebuah jaringan yang lebih luas, akan terdapat beragam perangkat komputer dan perangkat penghubung lainnya yang saling terhubung serta terjadi proses komunikasi dan transfer paket data di dalamnya.

A. Sifat-Sifat Dasar Jaringan Komputer

Jaringan komputer memiliki empat buah sifat dasar utama. Keempat sifat tersebut meliputi *scalability*, *resource sharing*, *connectivity*, dan *reliability*.

1. *Scalability* memiliki arti kemampuan untuk dapat diskalakan. Ini berarti jaringan komputer dapat diskalakan (diukur, disesuaikan) dengan kebutuhan pengguna jaringan komputer.

2. *Resource Sharing*

Resource sharing diartikan sebagai berbagi sumber daya. Dalam hal ini berarti bahwa jaringan komputer dapat digunakan untuk saling berbagi dan memakai secara bersama-sama segala sumber daya yang ada. Sumber daya meliputi seluruh perangkat keras komputer (*hardware*) dan perangkat lunak komputer (*software*).

3. *Connectivity*

Connectivity memiliki arti mudah terhubung dan dihubungkan. Hal ini berarti bahwa jaringan komputer memiliki sifat untuk mudah dihubungkan ke semua pengguna komputer dan semua pengguna komputer juga dapat dengan mudah terhubung ke dalam jaringan komputer yang tersedia. Untuk menciptakan hubungan ini terdapat sejumlah perangkat penghubung di dalamnya. Perangkat-perangkat tersebut antara lain berupa *Switch*, *Modem*, *Router*, *Hub*, perangkat *Wireless*, *Smartphone* dengan kemampuan *TEthering*, dan lain-lain.

4. *Reliability*

Reliability memiliki keandalan. Hal ini bahwa jaringan komputer memiliki kemampuan untuk dapat diandalkan di dalam jaringan komputer. Keandalan di sini dapat diartikan bahwa paket data yang dikirimkan oleh pengirim (*sender*) akan sampai dengan baik di sisi penerima (*reciever*). Keandalan yang semakin tinggi dan semakin baik pada jaringan komputer akan memberikan

kualitas layanan yang lebih baik bagi para pengguna jaringan komputer.

B. Jenis Jaringan Komputer

1. Local Area Network (LAN)

Jaringan *Local Area Network* (LAN) adalah suatu sistem jaringan dimana setiap komputer atau perangkat keras dan perangkat lunak di gabungkan agar dapat saling berkomunikasi (terintegrasi) dalam area kerja tertentu dengan menggunakan data dan program yang sama juga mempunyai kecepatan transfer data lebih cepat. Ruang lingkup LAN antar ruangan, gedung, kantor.

Beberapa keuntungan menggunakan LAN adalah :

- a. Dapat menghubungkan komputer dengan jumlah yang banyak.
- b. Akses antar komputer, baik untuk tukar menukar data atau yang lain, berlangsung cepat dan mudah.
- c. Dapat saling bertukar informasi dengan pengguna diluar area apabila terhubung dengan internet.
- d. Dapat membackup data pada komputer lain tanpa harus membongkar *harddisk*.
- e. Hemat waktu dan biaya dalam pengiriman paket data.

2. Metropolitan Area Network (MAN)

Metropolitan Area Network (MAN) merupakan pengembangan dari LAN. Jaringan ini terdiri dari beberapa jaringan LAN yang saling berhubungan. MAN biasanya digunakan oleh sebuah perusahaan jaringan komputer dalam satu kota dan mempunyai jangkauan antara 10 Km – 50 Km.

3. Wide Area Network (WAN)

Wide Area Network (WAN) merupakan jaringan komputer yang terdiri dari LAN dan MAN. WAN hanya menekankan pada fasilitas kecepatan akses transmisi sehingga memungkinkan seluruh komunikasi dapat berjalan secara lancar serta efisien.

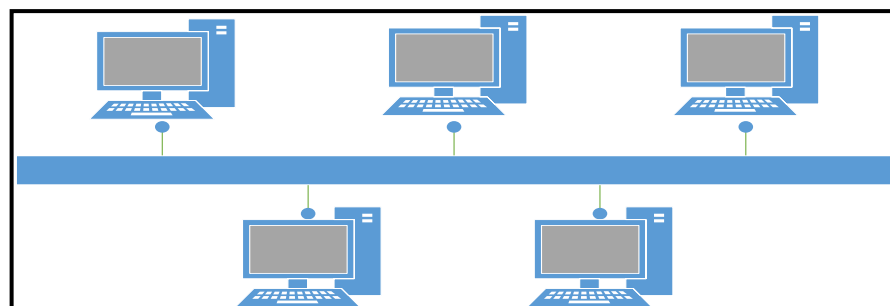
C. Topologi Jaringan

Menurut Syafrizal (2005:39), Topologi Jaringan atau arsitektur jaringan adalah gambaran perencanaan hubungan antar komputer dalam Local Area Network (LAN) yang umumnya menggunakan kabel (sebagai media transmisi), dengan konektor, *ethernet card*, dan perangkat pendukung lainnya.

Menurut Sofana (2012:113), LAN dapat dikembangkan dengan mudah dan mendukung kecepatan transfer data cukup tinggi. Ada 4 bentuk dasar LAN atau disebut juga topologi fisik LAN, antara lain sebagai berikut:

1. Topologi Bus

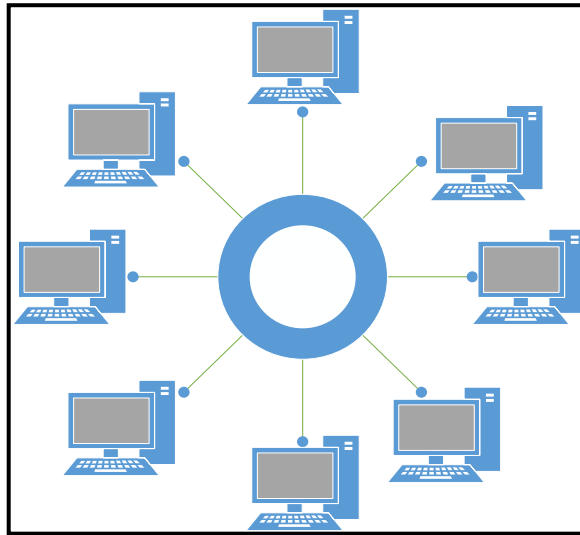
Topologi *bus* menggunakan sebuah kabel *backbone* dan semua *host* terhubung secara langsung pada kabel tersebut. Berikut ilustrasi dari topologi *bus*.



Gambar 2.2 Topologi Bus

2. Topologi Ring (Cincin)

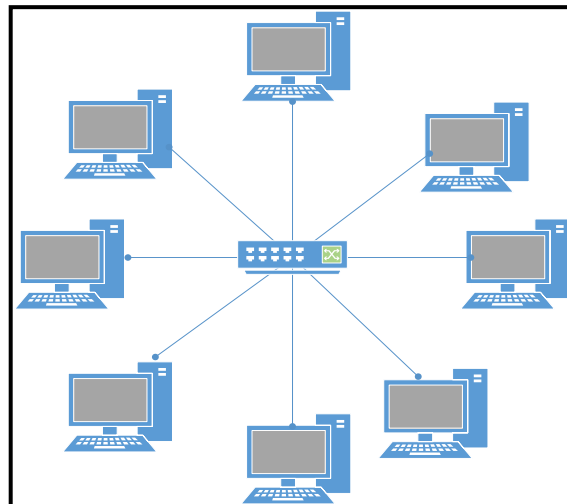
Topologi *ring* menghubungkan *host* dengan *host* lainnya hingga membentuk *ring* (lunakaran tertutup). Berikut ilustrasi dari topologi *ring*.



Gambar 2.3 Topologi Ring

3. Topologi Star

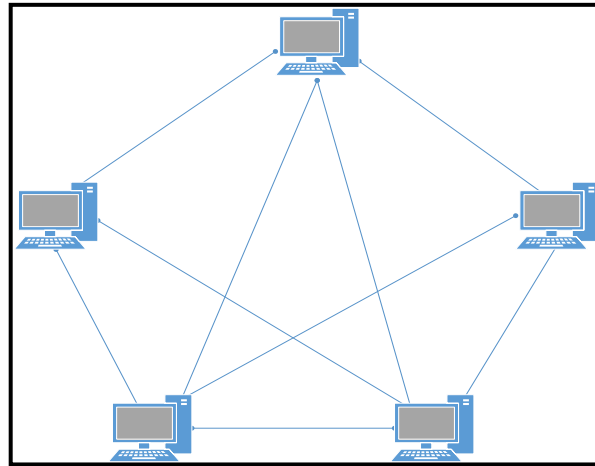
Topologi *star* menghubungkan semua komputer pada sentral atau konsentrator. Biasanya konsentrator adalah sebuah *hub* atau *switch*. Berikut ilustrasi dari topologi *star*.



Gambar 2.4 Topologi Star

4. Topologi Mesh atau Fully Mesh

Topologi *mesh* menghubungkan setiap komputer secara *point to point*. Artinya semua komputer akan saling terhubung satu – satu sehingga tidak dijumpai ada *link* yang putus. Berikut ilustrasi dari topologi *mesh*.



Gambar 2.5 Topologi Mesh

D. Protokol TCP/IP

Menurut Syafrizal (2005:96), Protokol TCP/IP adalah sekumpulan protokol yang terdapat di dalam jaringan komputer yang digunakan untuk berkomunikasi atau bertukar data antar komputer. TCP/IP merupakan protokol standar pada jaringan internet yang menghubungkan banyak komputer yang berbeda jenis maupun sistem operasi agar dapat berinteraksi satu sama lain.

Menurut Sofana (2012:245), TCP/IP memiliki karakteristik yang membedakannya dari protokol – protokol komunikasi yang lain, di antaranya:

1. Bersifat standar, terbuka dan tidak bergantung pada perangkat keras atau sistem operasi tertentu.

2. Bebas dari jaringan fisik tertentu, memungkinkan integrasi berbagai jenis jaringan.
3. Mempunyai skema pengalamatan yang umum bagi setiap *device* yang terhubung dengan jaringan.
4. Menyediakan berbagai layanan bagi user.

E. Model OSI

Menurut Syafrizal (2005:74), Model ini disebut OSI (*Open System Interconnection*) *Reference Model*, karena model ini ditujukan untuk pengkoneksian *open system*, yang dikembangkan oleh *International Organization for Standardization* (ISO) pada tahun 1984. *Open System* dapat diartikan sebagai suatu sistem yang terbuka untuk berkomunikasi dengan sistem-sistem lainnya. Untuk ringkasnya, kita akan menyebut model tersebut sebagai model OSI saja. OSI menggambarkan bagaimana informasi dari suatu *software* aplikasi pada sebuah komputer berpindah melewati sebuah media jaringan ke suatu *software* aplikasi di komputer lain.

Model OSI menyediakan secara konseptual kerangka kerja untuk komunikasi antar komputer, tetapi model ini bukan merupakan metode komunikasi. Sebenarnya komunikasi dapat terjadi karena menggunakan protokol komunikasi. Dalam konteks jaringan (komunikasi data), sebuah protokol adalah suatu aturan formal dan kesepakatan yang menentukan bagaimana komputer bertukar informasi melewati sebuah media jaringan. Sebuah protokol mengimplementasikan salah satu atau lebih lapisan-lapisan OSI.

OSI secara konseptual terbagi ke dalam 7 lapisan di mana masing-masing lapisan memiliki fungsi jaringan yang spesifik. Model ini diciptakan berdasarkan sebuah proposal yang dibuat oleh *the International Standards Organization* (ISO) sebagai langkah awal menuju standarisasi protokol internasional yang digunakan pada berbagai layer.

Ketujuh lapisan OSI dapat dibagi kedalam dua kategori, yaitu lapisan atas dan lapisan bawah seperti terlihat pada tabel 2.1 berikut.

Tabel 2.1 Lapisan Model OSI

<i>Application</i>	<i>Application</i>	Lapisan Atas
<i>Presentation</i>		
<i>Session</i>		
<i>Transport</i>	Data Transport	Lapisan Bawah
<i>Network</i>		
<i>Data Link</i>		
<i>Physical</i>		

Keterangan :

1. *Layer 7: Application Layer*

Application layer merupakan layer dimana terjadi interaksi antarmuka end user dengan aplikasi yang bekerja menggunakan fungsionalitas jaringan, melakukan pengaturan bagaimana aplikasi bekerja menggunakan resource jaringan, untuk kemudian memberika pesan ketika terjadi kesalahan. Beberapa service dan protokol yang berada di layer ini misalnya HTTP, FTP, SMTP, dll.

2. *Layer 6: Presentation Layer*

Layer ini bekerja dengan mentranslasikan format data yang hendak ditransmisikan oleh aplikasi melalui jaringan, ke dalam format yang bisa ditransmisikan oleh jaringan. Pada layer ini juga data akan di-enkripsi atau di-deskripsi.

3. *Layer 5: Session Layer*

Session layer akan mendefinisikan bagaimana koneksi dapat dibuat, dipelihara, atau dihancurkan. Di layer ini ada *protocol Name Recognition*, NFS & SMB.

4. *Layer 4: Transport Layer*

Layer ini akan melakukan pemecahan data ke dalam paket-paket data serta memberikan nomor urut pada paket-paket data tersebut sehingga dapat disusun kembali ketika sudah sampai pada sisi tujuan. Selain itu, pada layer ini, akan menentukan protokol yang akan digunakan untuk mentransmisi data, misalkan protokol TCP. Protokol ini akan mengirimkan paket data, sekaligus akan memastikan bahwa paket diterima dengan sukses (*acknowledgement*), dan mentransmisikan ulang terhadap paket-paket yang hilang atau rusak di tengah jalan.

5. *Layer 3: Network Layer*

Network layer akan membuat *header* untuk paket-paket yang berisi informasi IP, baik IP pengirim data maupun IP tujuan data. Pada kondisi tertentu, layer ini juga akan melakukan *routing* melalui *internetworking* dengan menggunakan *router* dan *switch layer-3*.

6. *Layer 2: Data-link Layer*

Befungsi untuk menentukan bagaimana bit-bit data dikelompokkan menjadi format yang disebut sebagai *frame*. Selain itu, pada level ini terjadi koreksi kesalahan, *flow control*, pengalamatan perangkat keras (seperti halnya *Media Access Control Address (MAC Address)*), dan menentukan bagaimana perangkat-perangkat jaringan seperti *hub*, *bridge*, *repeater*, dan *switch layer 2* beroperasi. Spesifikasi IEEE 802, membagi level ini menjadi dua level anak, yaitu lapisan *Logical Link Control (LLC)* dan lapisan *Media Access Control (MAC)*.

7. *Layer 1: Physical Layer*

Layer Physcal berkerja dengan mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti halnya *Ethernet* atau *Token Ring*), topologi jaringan dan pengkabelan. Selain itu, level ini juga mendefinisikan bagaimana *Network Interface Card (NIC)* dapat berinteraksi dengan media kabel atau radio.

2.9 Ubuntu Server

Menurut Husen dan Surbakti (2020:23), Linux Ubuntu *Server* adalah sistem operasi turunan dari Linux Ubuntu yang di desain khusus dengan kernel yang sesuai untuk beroperasi sebagai sistem operasi *server*. Kernel Linux Ubuntu *Server* di desain khusus untuk bisa bekerja dengan lebih dari satu proses (*multiprocessor*) dengan dukungan NUMA pada 100Hz in-ternal timer *frequency* dan menggunakan pennjadwalan *deadline I/O*.

Menurut febriantoro,dkk (21:2013) Ubuntu server adalah istilah yang dikenal dengan terminal. Terminal sering disebut command prompt atau shell. Dimasa lalu, hal ini adalah cara pengguna untuk berinteraksi dengan komputer, dan para penggunanya berpendapat bahwa penggunaan perintah melalui shell. Ubuntu suatu Operating System (OS) dari distro Linux. Sifatnya Open Source yang artinya semua orang bisa mengembangkan tanpa terpusat oleh suatu pengembang Ubuntu server memiliki kelebihan akan lebih cepat dibanding melalui aplikasi berbasis grafik dan hal ini masih berlaku sampai sekarang. Salah satu keunggulan lagi dari Ubuntu ini adalah adanya repository. Repository adalah tempat penyimpanan kumpulan software/aplikasi yang bias didownload untuk digunakan karena pada dasarnya setiap menginstall software di Linux dibutuhkan repository, maka Linux akan mencari software tersebut di repository. Jika telah ditemukan maka proses instalasi bisa dilanjutkan.

BAB III

METODOLOGI PENELITIAN

3.7 Subjek Penelitian

A. Tempat dan Waktu Penelitian

1. Tempat Penelitian

Penelitian dilaksanakan di laboratorium komputer UPT. Puskom Universitas Dehasen Bengkulu yang beralamatkan di Jl. Meranti Raya No. 32 Sawah Lebar Bengkulu.

2. Waktu Penelitian

Penelitian ini dilakukan dengan dua tahap yaitu:

a. Pra - Penelitian

Pra – penelitian ini dilakukan dari bulan Maret 2022 sampai dengan bulan Mei 2022.

b. Penelitian

Penelitian ini dilakukan dari bulan Mei 2022 sampai dengan bulan Juli 2022.

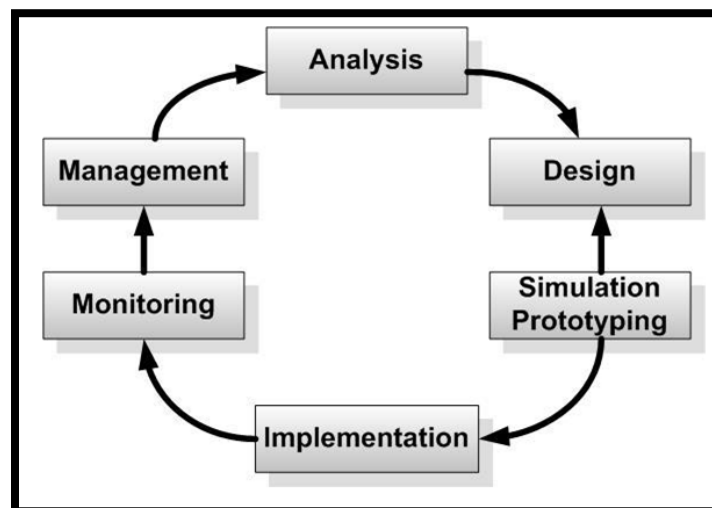
B. Struktur Organisasi

Struktur Organisasi merupakan kerangka kerja dimana didalamnya menggambarkan hubungan dan tanggung jawab setiap tingkat yang berada dalam Organisasi tersebut untuk melaksanakan demi tercapainya tujuan yang telah ditetapkan. Dengan demikian orang-orang tersebut mempunyai tugas, wewenang, dan tanggung jawab sesuai tugas masing-masing.

Struktur Organisasi sangatlah penting dalam suatu perusahaan atau instansi pemerintah. Karena dengan adanya struktur organisasi akan memperlihatkan dengan jelas kedudukan seseorang, sehingga setiap karyawan atau pegawai perusahaan atau instansi yang bersangkutan dapat mengetahui aktifitas dari perusahaan atau instansi dan dapat bekerja secara baik dari segi pembagian tugas maupun hal pelimpahan wewenang yang telah ditetapkan dalam struktur. Adapun struktur organisasi UPT. Puskom Universitas Dehasen Bengkulu terlampir.

3.8 Metode Penelitian

Metode penelitian yang digunakan adalah metode *Network Development Life Cycle* (NDLC). Dengan menggunakan metode penelitian NDLC ini diharapkan penulis dapat mendefinisikan siklus proses perancangan atau pengembangan suatu sistem jaringan komputer terutama dalam proses implementasi Pi Hole DNS *server* sebagai ads-blocker dan sistem *filtering* website pada jaringan hotspot. Berikut merupakan gambar dan tahapan dari metode penelitian NDLC, yaitu:



Gambar 3.1 NDLC Model

Dari Gambar 3.1 NDLC Model, dapat dijelaskan tahapan yang akan dilakukan seperti berikut ini:

1. Tahap *Analysis* (Analisis)

Tahapan analisis merupakan tahapan awal yang dilakukan dalam menganalisis kebutuhan, analisis permasalahan yang ada, analisis keinginan *user*, dan analisa topologi jaringan yang sudah ada, bisa dibilang tahapan ini adalah tahapan pengumpulan data yang dibutuhkan untuk perumusan masalah dalam menyelesaikan kendala yang ada. Dengan mengidentifikasi sistem yang sedang berjalan lalu mencoba untuk menganalisis suatu pengembangan sistem seperti apa yang akan diterapkan pada *database server* untuk menerapkan ketersediaan tinggi atau *high availability* menggunakan teknik *failover* dengan heartbeat.

2. Tahap *Design*

Pada tahapan ini, penulis akan membuat desain gambar topologi jaringan yang akan dibangun, desain akses data dan sebagainya dalam upaya implementasi *failover database server* menggunakan heartbeat.

3. Tahap *Simulation Prototype*

Pada tahapan ini, dilakukan simulasi pengembangan *database server* dengan menerapkan *failover database server* menggunakan heartbeat pada jaringan komputer. Hal ini dimaksudkan untuk melihat kinerja dari jaringan yang akan dibangun dan menjadi bahan presentasi dan *sharing* dengan pengembangan sistem jaringan.

4. Tahap *Implementation*

Pada tahapan ini, merupakan tahapan dimana implementasi *failover database server* menggunakan heartbeat. Pada tahapan ini juga akan terlihat bagaimana pengembangan yang akan dibangun serta apakah jaringan tersebut akan memberikan pengaruh terhadap sistem yang ada.

5. Tahap *Monitoring*

Tahapan *monitoring* merupakan tahapan penting agar jaringan dan komunikasi dapat berjalan sesuai dengan keinginan dan tujuan penulis pada tahap awal analisis.

6. Tahap *Management*

Pada tahap manajemen ini akan dilakukan beberapa langkah pengelolaan agar sistem yang telah dibangun dapat berjalan sesuai dengan yang diharapkan.

3.9 Instrumen Perangkat Lunak dan Perangkat Keras

Dalam melakukan penelitian ini, alat dan bahan yang digunakan meliputi perangkat lunak dan perangkat keras.

1. Perangkat Lunak (*Software*)

Adapun perangkat lunak (*software*) yang digunakan dalam penelitian ini dapat dilihat seperti berikut.

- a. Sistem Operasi Linux Ubuntu 20.04
- b. Apache
- c. PHP
- d. MySQL
- e. Heartbeat

f. HaProxy

g. Putty

2. Perangkat Keras (*Hardware*)

Adapun perangkat Keras (*hardware*) yang digunakan dalam penelitian ini dapat dilihat seperti berikut.

Tabel 3.1 Spesifikasi Perangkat Keras

No	Kebutuhan	Perangkat	Keterangan
1	3 unit PC <i>Server</i>	Processor	Intel(R) Pentium(R) CPU G620 @ 2.60GHz
		Memory	2 GB
		Harddisk	250 GB
		NIC	Fast Ethernet Card 10/100 MBps
2	2 Unit Komputer Client	Laptop	Intel Core i3
3	1 Unit Router	Mikrotik RB931- 2nD	a. Processor 650Mhz b. 3 port Fast Ethernet c. Build-in Wireless 2.4Ghz (802.11b/g/n) d. Antenna internal Dual-Chain 2 x 1.5dbi
4	1 Unit Switch	Tp-Link SF1008D Switch HUB 8Port	a. Interface : 8 10/100Mbps Ports, Auto-Negotiation, Auto-MDI/MDIX b. Buffer Size : 2Mb c. LED Indicator : Yes d. Power Supply : Output: 5.0VDC / 0.6A e. Adaptor : Yes, 220V, 50Hz D f. Data Rates : 10/100Mbps at Half Duplex, 20/200Mbps at Full Duplex Standards & g. Protocols : IEEE 802.3, IEEE 802.3u, IEEE 802.3x CSMA/CD

3.10 Metode Pengumpulan Data

Untuk memperoleh data yang diperlukan dalam penyusunan skripsi nanti penulis menggunakan beberapa metode dalam pengumpulan data yaitu:

A. Observasi

Merupakan metode pengumpulan data yang digunakan dengan cara melakukan pengamatan langsung pada jaringan yang ada di laboratorium komputer UPT. Puskom Universitas Dehasen Bengkulu.

B. Studi Pustaka

Merupakan metode pengumpulan data yang dilakukan dengan cara membaca buku-buku di perpustakaan kampus maupun perpustakaan daerah dan artikel di *internet* yang ada hubungannya dengan masalah yang dibahas dalam penelitian ini.

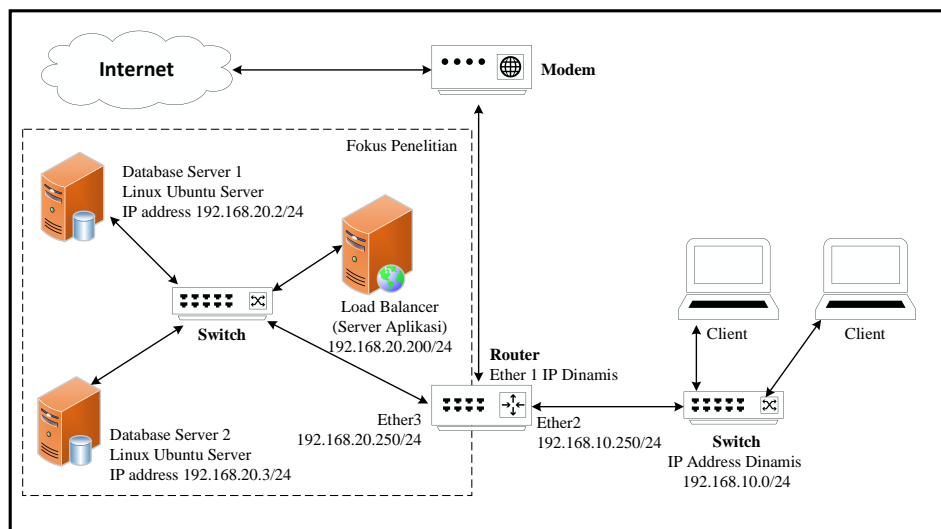
C. Studi Laboratorium

Data penelitian dikumpulkan dengan melakukan percobaan di laboratorium komputer UPT. Puskom Universitas Dehasen Bengkulu, mengenai cara perancangan *database server* dengan menerapkan teknik *failover*.

3.11 Metode Perancangan Sistem

A. Diagram Blok Sistem

Pada penelitian ini akan dilakukan pengembangan ataupun implementasi *failover* terhadap *database server* menggunakan heartbeat untuk meningkatkan ketersediaan tinggi atau *high availability* terhadap *server database* agar *request* data dari *client* dapat selalu dilayani. Adapun topologi yang akan digunakan adalah sebagai berikut.



Gambar 3.2 Diagram Blok Sistem Baru

Pada Gambar 3.2 tersebut dapat dilihat bahwa terdapat tambahan dua *database server* yang akan menjadi *failover cluster*, dimana pada *server* tersebut akan di *install* Linux Ubuntu *server* untuk sistem operasinya dan *software* heartbeat untuk *failover server* MySQL *server* sebagai *database server*. Selain itu akan ada *server* aplikasi yang akan menjadi *balancer* untuk aplikasi berbasis web yang akan menampilkan data yang ada pada *database server*.

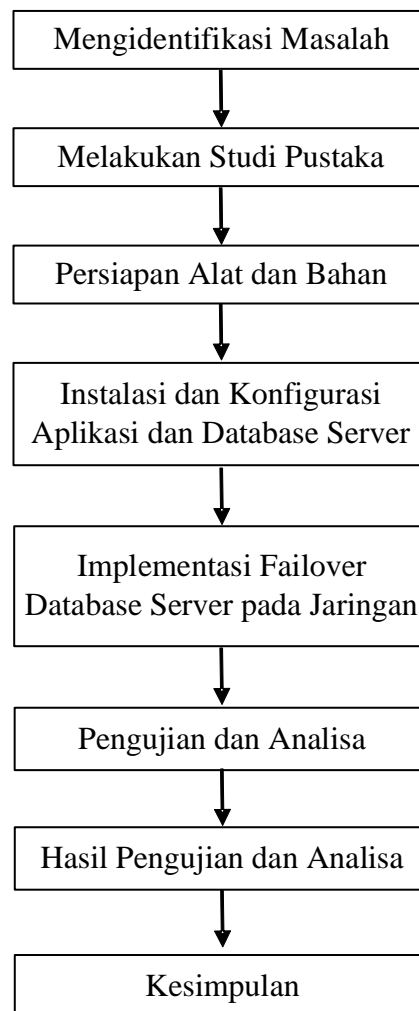
B. Prinsip Kerja Sistem

Prinsip kerja dari penelitian ini adalah menerapkan sistem perpindahan akses secara otomatis dari layanan database server untuk menerapkan ketersediaan tinggi atau *high availability*, sehingga request data dari client tidak terputus, dengan kata lain prinsip kerja dari *Failover* ini bertujuan untuk membantu menjaga akses *client* ke sumber daya di *server*, ketika *server* mengalami kegagalan fungsi *software*, atau kegagalan akses *server*. Sistem *Failover* akan bekerja untuk menghubungkan *server-server* yang menjadi anggota *cluster*-nya untuk

mengambil alih tugas *server* utama saat terjadi kegagalan fungsi dalam waktu tertentu

C. Rencana Kerja

Rencana kerja dari implementasi *Failover database server* menggunakan heartbeat adalah sebagai berikut.



Gambar 3.4 Rencana Kerja Sistem

Keterangan :

1. Mengidentifikasi Masalah

Identifikasi permasalahan dilakukan untuk menentukan masalah-masalah yang terjadi pada tempat penelitian, kemudian dirumuskan dan diberikan batasan-batasan permasalahan yang akan diteliti.

2. Melakukan Studi Pustaka

Studi pustaka dilakukan dengan cara membaca buku-buku di perpustakaan kampus maupun perpustakaan daerah dan artikel di *internet* yang ada hubungannya dengan masalah yang dibahas dalam penelitian ini.

3. Persiapan Alat dan Bahan

Adapun alat dan bahan yang harus disiapkan, antara lain sebagai berikut:

- a. Sistem Operasi Linux Ubuntu 20.04
- b. Apache
- c. PHP
- d. MySQL
- e. Heartbeat
- f. HaProxy
- g. Putty
- h. 3 Unit PC *Server*
- i. 2 Unit Laptop *Client*
- j. 1 Unit *Router*

k. 1 Unit *Switch* 8 Port

4. Instalasi dan Konfigurasi Aplikasi dan Database *Server*

Tahapan ini dilakukan untuk instalasi dan juga konfigurasi *server* yang akan digunakan untuk penerapan *failover database server* dalam jaringan komputer menggunakan heartbeat.

5. Implementasi *Failover Database Server* pada Jaringan

Tahapan ini dilakukan untuk menerapkan *Failover Database Server* pada jaringan.

6. Pengujian dan Analisa

Tahapan ini dilakukan untuk menguji sistem yang sudah diterapkan atau diimplementasikan pada jaringan. Apakah berjalan dengan baik ataupun sebaliknya.

7. Hasil Pengujian dan Analisa

Tahapan ini adalah tahapan yang dilakukan untuk menyimpulkan hasil dari pengujian yang dilakukan.

8. Kesimpulan

Pada tahapan ini adalah tahapan untuk menyimpulkan hasil dari penelitian yang telah dilakukan.

3.12 Rancangan Pengujian

Pengujian ini dilakukan dengan metode *blackbox*, yaitu sebuah metode yang digunakan untuk menemukan kesalahan dan mendemonstrasikan fungsional sistem saat dioperasikan, apakah *input* diterima dengan benar dan *output* yang dihasilkan telah sesuai

dengan yang diharapkan, sehingga dapat membuktikan kebenarannya.

Adapun rancangan pengujian dapat dilihat seperti Tabel 3.1 berikut.

Tabel 3.2 Pengujian dan Analisa

No	Jenis Pengujian	Kriteria	Hasil	Keterangan
1.	Pengujian Ketersediaan (Availability)	Pengujian dilakukan dengan menggunakan rumus availability.		
2.	Pengujian <i>Downtime</i> dan <i>Respon time</i>	Pengujian ini bertujuan untuk mengamati proses Failover dari kedua server		
3.	Pengujian Waktu Sinkronisasi data base	Pengujian ini dilakukan untuk mengetahui Waktu yang dibutuhkan MySQL server untuk sinkronisasi data.		

