



**LAPORAN PRAKTEK KERJA LAPANGAN**

**IMPLEMENTASI HIGH AVAILABILITY  
CLUSTER GUNA MENGURANGI DOWNTIME  
SERVER STUDI KASUS SERVER SINTASK**

Oleh:

ALMER HAFIZ WANDALAKSANA

NIM : 1308605011

Pembimbing:

I Wayan Supriana, S.Si., M.Cs

Program Studi Teknik Informatika

Jurusan Ilmu Komputer

Fakultas Matematika Dan Ilmu Pengetahuan Alam

Universitas Udayana

2016

**HALAMAN PENGESAHAN**  
**IMPLEMENTASI HIGH AVAILABILITY CLUSTER GUNA**  
**MENGURANGI DOWNTIME SERVER STUDI KASUS SERVER**  
**SINTASK**

Oleh:

Almer Hafiz Wandalaksana

NIM: 1308605011

Denpasar, 18 Mei 2017

Menyetujui,

Dosen Pembimbing



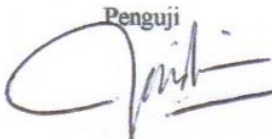
I Wayan Supriana, S.Si., M.Cs  
NIP.1984082920130122002

Pembimbing Lapangan



Tutde Suputrawan

Penguji



I Made Widiartha, S.Si., M.Kom  
NIP. 198212202008011008

Mengetahui,

Ketua Jurusan Ilmu Komputer  
FMIPA Universitas Udayana



Agus Muliartara, S.Kom, M.Kom.  
NIP.198006162005011001

## **KATA PENGANTAR**

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, Karena berkat rahmat dan karunia-Nya, laporan praktek kerja lapangan dengan judul “IMPLEMENTASI HIGH AVAILABILITY CLUSTER SERVER GUNA MENGURANGI DOWNTIME, STUDI KASUS SERVER SINTASK” ini dapat diselesaikan tepat pada waktu yang diberikan.

Selama melaksanakan praktek kerja lapangan dan dalam menyelesaikan laporan ini, penulis telah banyak menerima bimbingan, pengarahan, petunjuk, dan saran, serta fasilitas yang membantu hingga akhir penulisan laporan ini. Untuk itu, penulis mengucapkan terima kasih kepada yang terhormat :

1. Bapak Agus Muliantara, S.Kom, M.Kom. selaku ketua jurusan Jurusan Ilmu Komputer Fakultas MIPA Universitas Udayana yang telah memberikan pandangan, masukan, dan arahan selama penyusunan laporan ini.
2. Bapak I Wayan Supriana. selaku Dosen Pembimbing yang telah membantu dalam penyusunan laporan praktek kerja lapangan.
3. Semua pihak yang telah memberikan dorongan dan bantuan yang tidak mungkin disebutkan satu persatu.

Akhirnya penulis menyadari sepenuhnya bahwa proposal ini belum sempurna karena kemampuan yang ada pada penulis sangat terbatas dan semoga proposal ini dapat bermanfaat bagi pembaca.

Bukit Jimbaran, April 2017

Penulis

## **DAFTAR ISI**

HALAMAN PENGESAHAN .....	
KATA PENGANTAR.....	ii
DAFTAR ISI .....	iii
DAFTAR GAMBAR .....	vii
DAFTAR LAMPIRAN .....	ix
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	2
1.3 Manfaat.....	2
1.4 Waktu dan Pelaksanaan.....	2
BAB II GAMBARAN UMUM.....	3
2.1 Sejarah CV.SINTASK.....	3
2.2 Struktur Instansi Tempat PKL.....	3
2.3 Misi Tempat PKL.....	4
2.4 Kegiatan Instansi Tempat PKL .....	4
2.5 Keunggulan Sintask.....	4
2.6 Dampak Usaha.....	4

2.7	Target Pertumbuhan Pengguna.....	5
	<i>Gambar 2.2 Grafik Pertumbuhan Pengguna .....</i>	<i>5</i>
<b>BAB III KAJIAN PUSTAKA.....</b>		<b>6</b>
3.1	Jaringan Komputer .....	6
3.1.1	Manfaat Jaringan Komputer.....	6
3.1.2	Tipe Jaringan Komputer.....	7
3.1.3	Jenis Jaringan Komputer .....	8
3.1.4	Topologi Fisik Jaringan.....	10
3.1.5	Topologi Logika Jaringan .....	14
3.2	Konsep Dasar Protokol Internet .....	16
3.2.1	TCP.....	17
3.2.2	IP .....	18
3.3	Clustering Server .....	21
3.3.1	<i>Failover Clustering</i> .....	22
3.3.2	Load Balancing Clustering .....	23
3.3.3	Hubungan failover dan Loadbalancing.....	24
3.4	<i>Load Balancing</i> .....	24
3.4.1	Metode Load Balancing .....	26

3.4.2 Algoritma Load Balancing .....	27
<b>BAB IV PELAKSANAAN PKL .....</b>	<b>29</b>
4.1 Gagasan Multinode Server .....	29
4.2 Flowchart Implementasi Cluster Server .....	30
4.3 High Availability Cluster .....	31
4.4 Persiapan Implementasi <i>HACLUSTER</i> .....	32
4.4.1 Pengaturan Hostname dan <i>Ip Address</i> .....	32
4.4.2 Konfigurasi Iptables.....	33
4.4.3 Instalasi Komponen yang Dibutuhkan.....	33
4.5 <i>Konfigurasi Cluster</i> .....	34
4.5.1 Menambahkan <i>Password Default PCS</i> .....	34
4.5.2 Autentikasi <i>Node</i> .....	35
4.5.3 Konfigurasi Utama Pada Cluster .....	35
4.6 Konfigurasi <i>Virtual Ip</i> .....	36
4.7 Konfigurasi Webserver.....	36
4.8 <i>Error Checking</i> .....	37
4.9 Pengujian sistem.....	38
4.9.1 Pengujian <i>Virtual Ip</i> .....	38

4.9.2 Pengujian <i>Web Server</i> .....	39
4.9.3 Pengujian Failover Cluster.....	40
4.9.4 Pengujian Efisiensi Skalabilitas.....	41
4.9.5 Pengujian Latency Storage .....	43
<b>BAB V KESIMPULAN DAN SARAN</b> .....	<b>46</b>
5.1 Kesimpulan.....	46
5.2 Saran .....	46
<b>DAFTAR PUSTAKA</b> .....	<b>47</b>
<b>LAMPIRAN</b> .....	<b>49</b>

## **DAFTAR GAMBAR**

Gambar 2.1 Struktur Organisasi SINTASK .....	3
Gambar 2.2 Grafik Pertumbuhan Pengguna .....	5
Gambar 3.1 Skema Clustering Server .....	21
Gambar 3.2 Contoh Failover Clustering .....	22
Gambar 3.3 Contoh Load Balancing Cluster .....	23
Gambar 3.4 Contoh Penerapan Load Balancing .....	25
Gambar 4.1 Flowchart Implementasi Cluster Server .....	30
Gambar 4.2 Skema Node Clustering Server .....	31
Gambar 4.3 Pengaturan Iptables .....	33
Gambar 4.4 Proses Instalasi Komponen .....	34
Gambar 4.5 Merubah Password Node.....	34
Gambar 4.6 Pperintah Untuk Autentikasi Node.....	35
Gambar 4.7 Perintah Untuk Mematikan STONITH .....	36
Gambar 4.8 Perintah Untuk Membuat Resource Virtual Ip..	36
Gambar 4.9 Sintaks Memindahkan Listen Httpd .....	37
Gambar 4.10 Perintah Membuat Resource Webserver .....	37
Gambar 4.11 Hasil Error Checking.....	38



Gambar 4.12 Pengujian Virtual Ip .....	38
Gambar 4.13 Grafik Pengujian Ping Pada Cluster .....	39
Gambar 4.14 Pengujian Webserver .....	40
Gambar 4.15 Node1 dinonaktifkan .....	40
Gambar 4.16 Webserver Node2 .....	40
Gambar 4.17 Backend Server Clustering .....	41
Gambar 4.18 Skalabilitas Server .....	42
Gambar 4.19 Perbedaan metode penyimpanan .....	44

## DAFTAR LAMPIRAN

LAMPIRAN A .....	49
LAMPIRAN B .....	52

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kebutuhan dan gaya hidup masyarakat pada era digital saat ini menimbulkan perkembangan dalam berbagai bidang salah satunya teknologi. Pesatnya kemajuan teknologi semakin hari semakin memudahkan dan memanjakan manusia. Hal ini dapat kita lihat dari perkembangan teknologi komunikasi, interaksi yang terjadi dengan adanya bantuan teknologi menjadi semakin mudah, cepat, dan variatif. Ini membuat setiap orang dapat melakukan interaksi tanpa harus bertatap muka secara langsung. Berbagai bentuk layanan komunikasi berkembang untuk mewujudkan hal tersebut, dan saat ini kita menyebutnya sosial media. Sosial media merupakan suatu bentuk media komunikasi dimana kita dapat menjadikannya tempat bertukar informasi dan berkomunikasi yang dapat dilakukan dimana saja.

CV.SinTask merupakan badan usaha yang bergerak dibidang sosial media, dimana mempermudah penggunaanya dalam mengelola kebutuhan informasi dan komunikasi dalam suatu lingkup kerja. Sintask menyediakan layanan dengan membawakan inovasi dan melakukan kolaborasi antara sosial media yang bersifat fun dan produktif. Setiap pengguna bisa dengan mudah mengelola setiap tugas dan pekerjaan mereka sesuai *deadline* yang telah ditentukan sembari tetap menikmati hiburan yang telah disediakan Sintask.

Dalam memenuhi kebutuhan penggunaanya tentunya CV.SinTask membutuhkan sistem pendukung yang dapat diandalkan. Server yang menjadi infrastruktur utama yang mengolah interaksi user harus dapat bekerja dengan baik dan aksesibilitas tinggi. Tingkat kehandalan sebuah infrastruktur jaringan pada media sosial dapat diukur dengan lamanya *downtime* yang terjadi akibat gangguan infrastruktur. Jika proses pemulihan *downtime* yang ditimbulkan oleh gangguan memakan waktu lama, maka dapat dipastikan kenyamanan pengguna dalam menggunakan fitur dan layanan yang disediakan oleh Sintask.com akan terganggu.

*Downtime* yang dapat terjadi sewaktu waktu dikarenakan adanya gangguan yang disebabkan oleh berbagai faktor dapat dicegah dengan cara melakukan implementasi *cluster server* sehingga meminimalisir peluang terjadinya kegagalan yang mungkin terjadi pada *server* apabila terjadi masalah sehingga *server* gagal dalam melayani permintaan peng . Cluster server dibangun dengan menggunakan banyak server yang berjalan secara *redundant* dengan menyatukan layanan webserver dipadu dengan virtual ip. Cluster server ini dapat diakses melalui web (Sintask.com). Selain menggunakan URL, Akses terhadap cluster ini dapat melalui *static IP address*.

## 1.2 Tujuan

Adapun tujuan dari pembuatan dokumentasi ini adalah :

1. Melakukan implementasi high availability cluster Guna mengurangi downtime pada *server* Sintask.
2. Membantu tim pengembang sintask dalam mengembangkan infrastruktur jaringan server yang merupakan inti dalam menyediakan pelayanan dibidang media sosial.

## 1.3 Manfaat

Implementasi ini diharapkan dapat bermanfaat mengurangi *downtime server* jika terjadi kendala pada *server* sintask. Dengan demikian pelayanan yang diberikan akan dapat lebih optimal, dikarenakan teradapat backup server yang berjalan bersamaan dengan server utama sintask.

## 1.4 Waktu dan Pelaksanaan

Waktu pelaksanaan praktek kerja lapangan ini adalah selama kurang lebih dua bulan yang dimulai dari tanggal 6 februari 2017 sampai dengan 31 maret 2017. Pelaksanaan PKL disesuaikan dengan jam kerja dari instansi yaitu pukul 08.00 – 17.00 WITA dan pukul 16.00-06.00 WIITA.

Tempat pelaksanaan praktek kerja lapangan adalah di Gedung Animasi, Balai Diklat Industri yang bertempat di Jl. Wage Rudolf Supratman No.302, Kesiman Kertalangu, Denpasar Tim., Kota Denpasar, Bali 80237

## BAB II

### GAMBARAN UMUM

#### 2.1 Sejarah CV.SINTASK

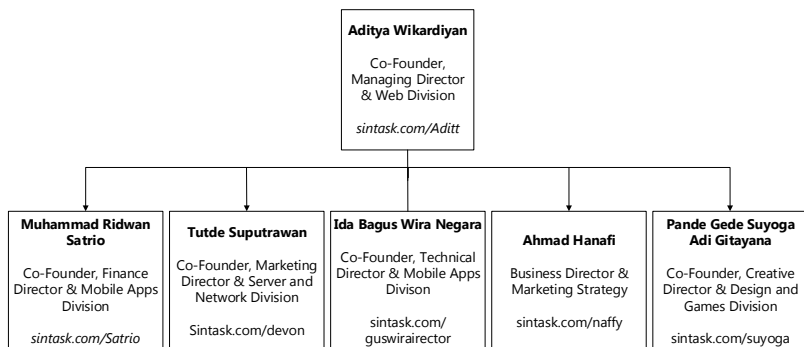
SinTask adalah Jejaring Sosial produktif Indonesia yang didirikan pada tahun 2015 oleh anak-anak muda Indonesia yang memiliki tujuan untuk meningkatkan produktifitas pengguna dengan memanfaatkan ketergantungan dalam menggunakan jejaring sosial.

SinTask memiliki fokus memberikan layanan task management dengan pendekatan yang mudah diterima yaitu media sosial. Dengan konsep Productive & Fun Social Network anda dapat bersenang-senang di dalam SinTask dengan bersosial media tanpa harus kehilangan produktivitas dalam memanajemen tugas.

SinTask menyediakan Fitur Produktif yang memudahkan individu maupun tim dapat dengan mudah memanajemen dan berkolaborasi dalam mengerjakan tugas. Kemudian Fitur Fun yang membuat anda dapat terhibur dengan menikmati media sosial yang ada di dalamnya.

#### 2.2 Struktur Instansi Tempat PKL

Struktur organisasi SINTASK ini dapat dilihat pada gambar di bawah ini.



**Gambar 2.1 Struktur Organisasi SINTASK**

### **2.3 Misi Tempat PKL**

CV. Sintask memiliki beberapa misi diantaranya sebagai berikut.

1. Menjadikan SinTask sebagai Jejaring Sosial di Indonesia yang mempunyai fokus meningkatkan produktifitas penggunaanya.
2. Menciptakan inovasi vital untuk mendukung kemandirian bangsa.

### **2.4 Kegiatan Instansi Tempat PKL**

Adapun kegiatan yang telah diikuti oleh CV.SinTask yaitu :

1. Teras Usaha Mahasiswa KOMPAS & BRI 2016 sebagai Local Champion
2. Event Jakarta Smart City Forum 2015 di Jakarta
3. Event Pameran Produksi Indonesia 2015 di Surabaya
4. Event Futurepreneur Roadshow 2015 di BCIC Denpasar
5. Expo Kewirausahaan Muda Indonesia 2016 di Universitas Brawijaya

### **2.5 Keunggulan Sintask**

SinTask merupakan jejaring sosial yang memudahkan anda untuk bekerja dan bersosialisasi di saat yang sama serta mendapatkan hiburan secara seimbang. Adapun keunggulan sintask dibandingkan dengan media sosial yang lainnya diantaranya :

1. Pengguna dapat melakukan aktifitas bersosial media tanpa khawatir melupakan kewajibannya dalam mengerjakan tugas.
2. Bentuk produktivitas yang dihadirkan adalah Pengelolaan Tugas (Task Management).
3. Bentuk hiburan yang dihadirkan adalah interaksi Social Media.

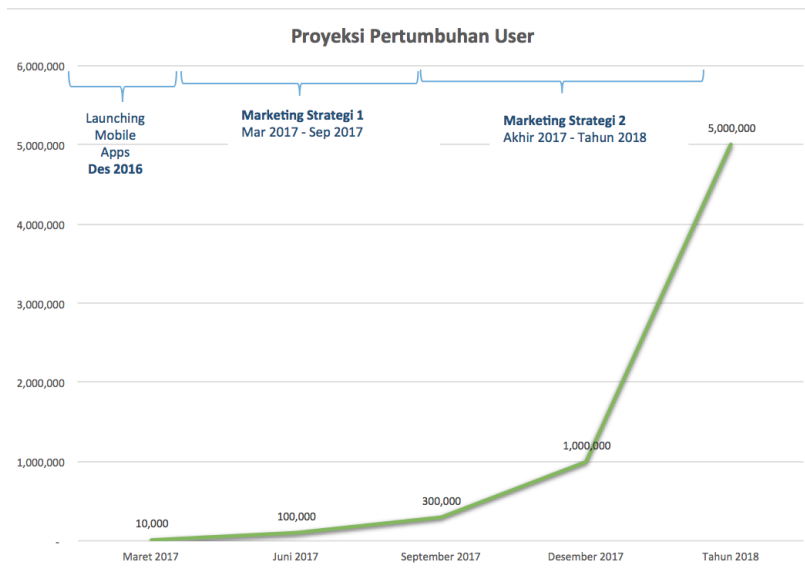
### **2.6 Dampak Usaha**

1. Mengurangi penggunaan bandwidth internasional karena server SinTask berada di Indonesia.
2. Memberikan mindshare kepada pengguna bahwa tugas bukanlah hal yang membosankan ataupun menakutkan, melainkan hal yang sangat menyenangkan.

3. Membuat pengguna menjadi lebih bijak dalam manajemen waktu ketika sedang bersosial media.
4. Menyeimbangkan tingkat produktifitas dan hiburan pengguna.

## 2.7 Target Pertumbuhan Pengguna

Dengan melihat pengguna media sosial yang ada pada saat ini, CV.Sintask optimis pada tahun 2018 pengguna layanan Sintask.com mencapai 5.000.000 pengguna dengan rincian sebagai berikut.



**Gambar 2.2 Grafik Pertumbuhan Pengguna**

## **BAB III**

### **KAJIAN PUSTAKA**

#### **3.1 Jaringan Komputer**

Jaringan Komputer merupakan kumpulan dari perangkat keras dan lunak di dalam suatu sistem yang memiliki aturan tertentu untuk mengatur seluruh anggotanya dalam melakukan aktivitas komunikasi. (Tanaenbaum, 2002) Perangkat yang terhubung dengan jaringan disebut juga sebagai node. Hal ini memungkinkan pengguna dapat bertukar dokumen dan data, mencetak pada printer yang sama, dan menggunakan sumber daya jaringan (hardware dan software) ada.

##### **3.1.1 Manfaat Jaringan Komputer**

###### **1. Resource Sharing**

Penggunaan jaringan komputer (*network*) memungkinkan dapat menggunakan sumberdaya yang secara bersama-sama. Misalnya seorang pengguna yang berada di 100 Km jauhnya dari suatu data, tidak mendapatkan kesulitan dalam menggunakan data tersebut dan seolah olah data tersebut berada di dekat. Hal ini sering diartikan bahwa jaringan komputer mengatasi masalah jarak.

###### **2. Integrasi Data**

Proses pertukaran data dengan menggunakan jaringan komputer memungkinkan pengolahan data dapat dilakukan dan didistribusikan ke beberapa komputer. Proses ini menyebabkan terjadinya integrasi data yang dapat diakses secara tepat, cepat dan akurat

###### **3. Keamanan Data**

Tidak dipungkiri bahwa adanya jaringan komputer dapat menyebabkan penyebaran virus secara merata ke semua komputer. Hal ini dapat diatasi dengan menggunakan antivirus terbaru dan pencegahan masuknya disket di sebarang komputer. Adanya jaringan komputer memberikan keamanan bagi pemakai komputer karena hanya pemakai tertentu saja yang dapat menggunakan komputer. Hal



ini akan mencegah penggunaan komputer oleh orang lain yang dapat mengganggu keamanan data dalam komputer.

#### 4. Efisiensi Sumber Daya

Adanya sharing resource atau berbagi perangkat keras dapat menghemat biaya pengadaan perangkat keras (hardware). Misalnya, suatu perusahaan tidak perlu membeli 10 printer untuk 10 komputer.

#### 5. Sarana Komunikasi.

Dengan adanya dukungan jaringan komputer, komunikasi dapat dikerjakan dengan lebih cepat. Para pengguna komputer dapat mengirimkan surat elektronik (*e-mail*) dengan mudah.

### 3.1.2 Tipe Jaringan Komputer

Dalam jaringan komputer, terdapat tiga peranan yang dapat dijalankan oleh komputer-komputer di dalam LAN (Local Area Network). Peran pertama bisa menjadi client, yaitu hanya sebagai pengguna saja tapi tidak menyediakan sumber daya jaringan untuk dishare dibagi dan dipakai oleh anggota jaringan lain. Peran kedua bisa menjadi peer, yaitu sebagai klien yang menggunakan sekaligus menyediakan sumber daya jaringan yang disebut sebagai peer-to-peer. Peran terakhir adalah bisa menjadi server yang menyediakan sumber daya jaringan. Berdasarkan tiga peranan diatas, selanjutnya jaringan komputer terbagi atas 3 bagian yaitu:

#### 1. Jaringan Peer to Peer

Jaringan peer to peer adalah jenis jaringan komputer dimana setiap komputer bisa menjadi server sekaligus client. Setiap komputer dapat menerima dan memberikan akses dari/ke komputer lain. Peer to peer banyak diimplementasikan pada LAN, karena cukup sulit mengawasi security pada jaringan peer to peer manakala pengguna komputer sudah sangat banyak.

#### 2. Jaringan Berbasis Server dan Client-Server

Didefinisikan dengan kehadiran server didalam suatu jaringan yang menyediakan mekanisme pengamanan dan pengelolaan jaringan tersebut. Jaringan ini terdiri dari satu atau lebih server dan banyak

klien yang biasa disebut sebagai komputer frint-end, meminta layanan diantaranya penyimpanan dan pencetakan data ke printer jaringan, sedangkan server yang biasa disebut sebagai komputer back-end menyampaikan permintaan tersebut ke tujuan yang tepat. Jaringan peer- to-peer. Secara sederhana jaringan ini digambarkan, setiap komputer pada jaringan peer-to-peer berfungsi sebagai client dan server sekaligus.

### 3. Jaringan *Hybrid*

Jaringan *Hybrid* adalah jaringan komputer yang memiliki semua yang terdapat pada dua tipe jaringan *client server* dan *peer-to-peer*. Ini berarti pengguna dalam jaringan ini dapat mengakses sumber daya yang dishare atau dibagi pakai oleh jaringan *peer-to-peer*, sedangkan pada saat yang bersamaan juga dapat memanfaatkan sumber daya yang disediakan oleh komputer server.

#### 3.1.3 Jenis Jaringan Komputer

Berdasarkan luas areanya jaringan komputer dapat dibagi menjadi empat, yaitu PAN (*Personal Area Network*), LAN (*Local Area Network*), MAN (*Metropolitan Area Network*), WAN (*Wide Area Network*). (Iwan Sofana, 2012). Berikut Uraiannya:

##### 1. Personal Area Network (PAN)

merupakan jaringan komputer yang dibentuk oleh beberapa buah komputer atau antara komputer dengan peralatan non-komputer (seperti: printer, mesin *fax*, telepon seluler, PDA, handphone). Sebuah PAN dapat dibangun menggunakan teknologi *wire* dan *wireless network*. Teknologi wire PAN biasanya mengandalkan perangkat USB dan FireWire. Sedangkan *wireless* PAN mengandalkan teknologi *Bluetooth*, *WIFI*, dan *Infrared*. Saat ini, *wireless* PAN (WPAN) yang menggunakan *Bluetooth* lebih disukai pengguna. Sebuah WPAN dapat dibangun dengan cepat berkat kehadiran perangkat *Bluetooth*

## 2. Local Area Network (LAN)

*Local Area Network* (LAN) merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer. LAN biasanya digunakan untuk menghubungkan komputer-komputer pribadi dan *workstations* dalam kantor perusahaan atau pabrik-pabrik untuk memakai bersama *resource* (misalnya, *printer*, *scanner*) dan saling bertukar informasi. LAN dapat dibedakan dari jenis jaringan lainnya berdasarkan tiga karakteristik, yaitu ukuran teknologi, topologi dan transmisinya. LAN seringkali menggunakan teknologi transmisi kabel tunggal. LAN tradisional bekerja pada kecepatan mulai 10 sampai 100 Mbps (mega bit/detik) dengan delay rendah (puluhan mikro second) dan mempunyai factor kesalahan yang kecil. LAN-LAN modern dapat bekerja pada kecepatan yang lebih tinggi, hingga ratusan mega bit/detik.

### Keuntungan dari *Local Area Network* (LAN)

- a. Pertukaran *file* dapat dilakukan dengan mudah (*file sharing*).
- b. Pemakaian printer dapat dilakukan oleh semua *client* (*printer sharing*). *File-file* data bisa disimpan pada *server*, dan dapat diakses dari semua *client* menurut otoritas dari semua karyawan, yang dapat dibuat menurut otoritas sekuritas dari semua karyawan, yang dapat dibuat menurut struktur organisasi perusahaan sehingga keamanan data terjamin.
- c. Proses *backup* data menjadi lebih mudah dan cepat.
- d. Resiko kehilangan data oleh virus komputer menjadi sangat kecil.
- e. Komunikasi antar karyawan dapat dilakukan dengan menggunakan *e-mail* dan chat.
- f. Bila salah satu *client/server* terhubung dengan modem, maka sebagian atau semua komputer pada jaringan LAN dapat mengakses ke jaringan *internet* melalui satu modem.

## 3. Metropolitan Area Network (MAN)

Jaringan ini lebih luas dibandingkan jaringan LAN dan menjangkau antar wilayah dalam satu provinsi. Jaringan MAN

menghubungkan beberapa jaringan kecil yang ada, seperti LAN yang menuju lingkungan *area* yang lebih besar. Misalnya, beberapa bank yang memiliki jaringan komputer di setiap cabangnya dapat berhubungan satu sama lain sehingga nasabah dapat melakukan transaksi di cabang maupun dalam provinsi yang sama

#### 4. Wide Area Network

Jaringan ini mencakup area yang luas dan mampu menjangkau batas provinsi bahkan sampai negara yang ada dibelahan bumi lain. Jaringan WAN mampu menghubungkan satu komputer dengan komputer lain dengan menggunakan satelit atau kabel bawah laut. Topologi yang digunakan WAN yaitu topologi tak menentu sesuai dengan apa yang akan di gunakan.

### 3.1.4 Topologi Fisik Jaringan

Sebuah jaringan komputer dibangun menggunakan suatu topologi jaringan. Tidak semua topologi jaringan sesuai untuk digunakan dalam sebuah jaringan komputer. Hal itu disebabkan oleh sumber daya yang akan digunakan untuk membangun jaringan. Oleh sebab itu seorang administrator jaringan harus cermat dalam memilih topologi yang cocok untuk jaringan yang akan di buatnya. Berikut adalah beberapa jenis topologi jaringan yang umum digunakan:

#### 1. Topologi Bus

Topologi Bus (topologi backbone) adalah topologi jaringan dengan membentangkan kabel (coaxial) memanjang dengan kedua ujungnya ditutup dimana sepanjang kabel terdapat node-node kemudian perangkat jaringan dan komputer-komputer dihubungkan pada kabel tersebut menggunakan T-Connector.

Ciri-ciri Topologi Bus

- a. Teknologi lama yang umum digunakan
- b. Tidak butuh peralatan aktif dalam menghubungkan komputer
- c. Menggunakan konektor BNC tipe T
- d. Pada ujung kabel dipasang konektor 50ohm
- e. Diperlukan repeater untuk jarak yang cukup jauh
- f. Discontinue Support

### Kelebihan Topologi Bus

- a. Hemat Kabel
- b. Layout kabel sangat sederhana
- c. Biaya instalasi relatif lebih murah
- d. Penambahan workstation baru mudah dilakukan tanpa mengganggu workstation yang lain

### Kekurangan Topologi Bus

- a. Sulit melakukan pelacakan masalah
- b. Signal melewati kabel dalam dua arah dan mungkin terjadi collision (tabrakan pengiriman data)
- c. Problem terbesar pada saat kabel putus. Jika salah satu segment kabel putus, maka seluruh jaringan akan terhenti dan komputer tidak dapat saling berkomunikasi.

## 2. Topologi Star

Disebut topologi *star* karena bentuknya seperti bintang, sebuah alat yang disebut *concentrator* dapat berupa *hub* atau *switch* menjadi pusat, dimana semua perangkat pada jaringan dihubungkan ke *concentrator* ini.

### Kelebihan topologi star :

- a. Karena setiap komponen dihubungkan langsung ke simpul pusat maka pengelolaan menjadi mudah.
- b. Kegagalan komunikasi mudah ditelusuri.
- c. Kegagalan pada satu komponen/terminal tidak mempengaruhi komunikasi terminal lain.
- d. Kontrol terpusat sehingga memudahkan dalam deteksi dan isolasi kesalahan serta memudahkan pengelolaan jaringan.

### Kekurangan topologi star :

- a. Kegagalan pusat kontrol (simpul pusat) memutuskan semua komunikasi.
- b. Boros dalam penggunaan kabel.

- c. Kondisi switch harus tetap dalam kondisi baik, kerusakan switch menyebabkan lumpuhnya seluruh link dalam jaringan sehingga computer tidak dapat saling berkomunikasi.

### 3. Topologi Ring

Disebut topologi ring karena bentuknya seperti cincin yang melingkar. Semua komputer pada jaringan di hubungkan pada sebuah cincin. Cincin ini hampir sama kegunaannya dengan concentrator pada topologi star yang menjadi pusat berkumpulnya ujung kabel dari setiap komputer yang terhubung.

Kelebihan topologi ring:

- a. Dapat melayani aliran lalu lintas data yang padat.
- b. Aliran data mengalir lebih cepat karena dapat melayani data dari kiri atau kanan dari server.
- c. Trasmisi data yang relatif sederhana seperti perjalanan paket data dalam satu arah saja.

Kekurangan topologi ring :

- a. Kerusakan pada salah satu media pengirim/terminal dapat melumpuhkan kerja seluruh jaringan.
- b. Paket data harus melewati setiap komputer antara pengirim dan penerima, yang menyebabkan menjadi lebih lambat.
- c. Pengembangan jaringan menjadi lebih kaku karena penambahan terminal atau node menjadi lebih sulit bilaport sudah habis.

### 4. Topologi Tree

Topologi pohon adalah pengembangan atau generalisasi topologi *bus*. Media transmisi yaitu berupa satu kabel yang bercabang namun *loop* tidak tertutup

Kelebihan topologi tree :

- a. Memungkinkan untuk memiliki jaringan point to point.
- b. Mengatasi keterbatasan pada topologi star, yang memiliki kekurangan dalam titik koneksi hub.

- c. Topologi tree membagi seluruh jaringan menjadi bagian yang lebih mudah diatur.
- d. Topologi tree ini memiliki keunggulan lebih mampu menjangkau jarak yang lebih jauh dengan mengaktifkan fungsi Repeater yang dimiliki oleh HUB.

Kekurangan topologi tree :

- a. Karena bercabang maka diperlukan cara untuk menunjukkan kemana data dikirim, atau kepada siapa transmisi data ditujukan.
  - b. Perlu suatu mekanisme untuk mengatur transmisi dari terminal terminal dalam jaringan.
  - c. Kabel yang digunakan menjadi lebih banyak sehingga diperlukan perencanaan yang matang dalam pengaturannya, termasuk tata letak ruangan.
  - d. HUB menjadi elemen penting.
5. Topologi Mesh
- Topologi Mesh adalah topologi yang tidak memiliki aturan dalam koneksi. Karena tidak teratur sehingga kegagalan komunikasi menjadi sulit dideteksi, dan ada kemungkinan boros dalam penggunaan media transmisi. Setiap perangkat terhubung secara langsung ke perangkat lainnya yang ada di dalam jaringan. Sehingga, pada topologi mesh setiap perangkat dapat berkomunikasi langsung dengan perangkat yang dituju.

Kelebihan topologi mesh :

- a. Dapat berkomunikasi langsung dengan perangkat tujuan.
- b. Data dapat di kirim langsung ke komputer tujuan tanpa harus melalui komputer lainnya lebih cepat. Satu link dipergunakan hanya untuk berkomunikasi dengan komputer yang di tuju.
- c. Memiliki sifat Robust, yaitu Apabila terjadi gangguan pada koneksi komputer A dengan komputer B karena rusaknya kabel koneksi (links) antara A dan B, maka gangguan itu tidak

akan mengganggu koneksi komputer A dengan komputer lainnya.

- d. Mudah dalam proses identifikasi permasalahan pada saat terjadi kerusakan koneksi antar komputer.

Kekurangan topologi mesh :

- a. Setiap perangkat harus memiliki I/O port. Butuh banyak kabel dan menyebabkan butuh banyak biaya.
- b. Instalasi dan konfigurasi lebih sulit karena komputer yang satu dengan yang lain harus terkoneksi secara langsung.
- c. Biaya yang besar untuk memelihara hubungan yang berlebih

### 3.1.5 Topologi Logika Jaringan

Topologi Logic berasal dari kata “ *Logic* “ yang berarti adalah suatu gambaran bagaimana hubungan yang terjadi antar masing-masing komputer dalam jaringan yang tidak dapat kita lihat, tetapi dapat kita mengalaminya ( merasakan ). Pengertian secara umum Topologi Logic merupakan topologi yang menggambarkan hubungan secara logika yang terjadi pada masing-masing komputer dalam jaringan. Dalam Topologi Logic terbagi dalam beberapa bentuk arsitektur yang telah ada diantaranya adalah :

#### 1. Ethernet

Ethernet menggunakan metode akses yang disebut CSMA/CD ( *Carrier Sense Multiple Access/Collision Detection* ). Sistem ini dapat memperhatikan setiap komputer kedalam kabel dari network sebelum mengirimkan data ke dalamnya. Jika dalam jaringan tidak ada aktifitas, komputer akan mentransmisikan data. Jika ada transmisi lain di dalam kabel, komputer akan menunggu dan akan mencoba kembali transmisi ketika jaringan telah kosong.

Jika ada dua buah komputer melakukan transmisi pada saat bersamaan, maka komputer akan mundur dan akan menunggu kesempatan secara acak untuk mentransmisikan data kembali. Metode ini disebut dengan koalisi, yang tidak akan berpengaruh pada kecepatan transmisi dari network. Ethernet dapat digunakan pada



model jaringan Garis lurus , Bintang, atau Pohon. Data dapat ditransmisikan melewati kabel twisted pair, koaksial, ataupun kabel fiber optik pada kecepatan 10 Mbps - 100Mbps dan terus berkembang sampai 1Gbps. Cara kerja arsitektur ini memakai metoda CSMA/CD (*Carrier Sence Multiple Acces/collision detection*). Bilamana suatu node mengirimkan paket melewati jaringan, maka node tersebut akan mengecek terlebih dahulu apakah jaringan sedang mengirimkan paket data atau tidak. jika jaringan sedang kosong, maka node akan mengirimkan paket data. Jika ternyata ada paket data lain, pada saat node akan mengirimkan data, maka akan terjadikollision. Bila hal ini terjadi maka jaringan dan node akan berhenti mengirimkan paket data, kemudian node dan jaringan. Secara sederhana dapat digambarkan yaitu suatu host yang mengirimkan data kepada seluruh host lain pada media jaringan

## 2. Token Ring

Token Ring dikembangkan oleh IBM pada pertengahan tahun 1980. Hubungan komputer pada token berbentuk seperti cincin. Sebuah Sinyal token bergerak berputar seperti lingkaran pada sebuah jaringan dari satu komputer menuju ke komputer yang lain. Token Ring membutuhkan model jaringan Bintang (STAR) dengan menggunakan kabel twisted pair atau kabel fiber optik yang dapat melakukan kecepatan transmisi 4 Mbps atau 16 Mbps. Sejalan dengan perkembangan Ethernet, penggunaan Token Ring makin berkurang sampai sekarang. Cara kerja Token Ring, sebuah token bebas mengalir dalam jaringan, jika satu node ingin mengirimkan paket data, maka paket data yang akan dikirimkan ditempelkan pada token, pada waktu token berisi data, node lain tidak dapat mengirimkan data. *Token passing* digunakan dalam arsitektur ini untuk menghindari *collision*.

Data dalam jaringan dikirim oleh masing-masing komputer yang kemudian berjalan melingkar ke komputer-komputer yang lain untuk kemudian data tersebut akan diambil oleh komputer yang dituju atau yang membutuhkan. Pola transmisi ini tetap berlaku meskipun topologi menggunakan STAR. Kelebihan token ring menggunakan token passing guna menghindari tabrakan data kecepatan mencapai 16 Mbps, menggunakan kabel fiber optik.

Adapun kekurangan dari token ring jika terdapat gangguan di node maka jaringan akan terganggu

### 3. FDDI (Fiber Distributed Data Interface)

Fiber Distributed Data Interface (FDDI) adalah sebuah jaringan yang menghubungkan antara dua atau lebih jaringan bahkan pada jarak yang jauh. Metode yang digunakan oleh FDDI adalah model token ring. FDDI menggunakan dua buah topologi ring secara fisik. Proses transmisi biasanya menggunakan satu buah ring, namun jika ada masalah ditemukan akan secara otomatis menggunakan ring yang kedua.

FDDI menggunakan kabel fiber optik yang bekerja berdasarkan 2 buah ring konsentris dengan kecepatan 100Mbps. Salah satu ring bisa berfungsi sebagai backup apabila ring yang lainnya atau node (komputer) lain terputus atau tidak beroperasi. Jaringan dengan arsitektur ini memerlukan biaya yang cukup mahal, sehingga kurang cocok untuk membangun jaringan komputer yang sederhana seperti di rumah atau di kantor-kantor kecil. Kecepatan FDDI dengan menggunakan fiber optik kabel mencapai 100 Mbps. FDDI dapat menghubungkan sampai 500 terminal dengan jarak maksimum 2 km.

### 4. ATM (Asynchronous Transfer Mode)

ATM (Asynchronous Transfer Mode) yaitu sebuah jaringan yang mentransmisikan pada kecepatan 155 Mbps atau lebih. ATM mentransmisikan data kedalam satu paket, sedangkan yang lain mentransfer pada besar-kecilnya paket. ATM mendukung variasi media seperti video, CD-audio, dan gambar. ATM bekerja pada model topologi Bintang dengan menggunakan kabel fiber optik ataupun kabel twisted pair. ATM pada umumnya digunakan untuk menghubungkan dua atau lebih LAN. ATM juga banyak dipakai oleh Internet Service Providers (ISP) untuk meningkatkan kecepatan akses Internet untuk klien mereka.

## 3.2 Konsep Dasar Protokol Internet

Protokol dapat di ibaratkan sebagai penerjemah dua orang yang berbeda bahasa ingin berkomunikasi. Protokol internet yang

pertama kali dirancang pada awal tahun 1980-an. Akan tetapi pada saat itu, protokol tersebut hanya digunakan untuk menghubungkan beberapa node saja. Baru pada awal tahun 1990-an mulai disadari bahwa internet mulai tumbuh ke seluruh dunia dengan pesat. Sehingga banyak bermunculan protokol internet. Sehingga disadari bahwa dibutuhkan sebuah protokol internet yang standar, yaitu OSI (Open System Interconnection). Tetapi pada perkembangannya, TCP/IP menjadi standar *de facto* yaitu standar yang diterima karena pemakaiannya secara sendirinya semakin berkembang. Kelebihan dari Ethernet diantaranya adalah cukup sederhana, mudah dalam aplikasinya, kecepatan yang up to 10Gbps. Diantara kelebihan tersebut masih terdapat kekurangan seperti sering terjadi *collision* data dan *degradation over time*.

### 3.2.1 TCP

Pada saat mengirim atau menerima data, data yang ditransmisikan menjadi potongan potongan kecil yang disebut paket. Masing-masing paket ini berisi alamat internet pengirim dan penerima. Setiap paket yang dikirim akan melewati *gateway* dari jaringan yang termurupakan bagian yang menghubungkan antar jaringan. Alamat dari paket yang ditransmisikan sudah didefinisikan namun gateway hanya akan mengirimkan paket tersebut meneruskan ke gateway terdekatnya, hingga sebuah gateway mengenali paket tersebut berada pada domain terdekatnya.

Paket yang dikirimkan dapat melewati rute yang berbeda beda, dengan demikian *sequence* saat paket diterima bisa jadi tidak berurutan oleh karena itu setiap paket terdapat *sequence number* yang diberikan oleh sebuah protocol yaitu *Transmission Control Protocol (TCP)*. Transmission Control Protocol atau yang sering kali disingkat menjadi TCP berfungsi untuk melakukan transmisi data per-segmen (paket data dipecah dalam jumlah yang sesuai dengan besaran paket kemudian dikirim satu persatu hingga selesai). Agar pengiriman data sampai dengan baik, maka pada setiap packet pengiriman, TCP akan menyertakan nomor

seri (sequence number). Adapun komputer tujuan yang menerima paket tersebut harus mengirim balik sebuah sinyal acknowledge dalam satu periode yang ditentukan. Bila pada waktunya komputer tujuan belum juga memberikan acknowledge, maka terjadi time out yang menandakan pengiriman packet gagal dan harus diulang kembali. Model protokol TCP disebut sebagai connection oriented protocol.

### 3.2.2 IP

IP (Internet Protocol) atau alamat IP dapat disebut dengan kode pengenalan komputer pada jaringan merupakan komponen vital pada internet, karena tanpa alamat IP seseorang tidak akan dapat terhubung ke internet. Penggunaan alamat IP dikoordinasi oleh lembaga sentral internet yang dikenal dengan IANA, salah satunya adalah NIC (Network Information Center).

Adanya IP Address merupakan konsekuensi dari penerapan Internet Protocol untuk mengintegrasikan jaringan komputer Internet di dunia. Seluruh host (komputer) yang terhubung ke Internet dan ingin berkomunikasi memakai TCP/IP harus memiliki IP Address sebagai alat pengenalan host pada network. Secara logika, Internet merupakan suatu network besar yang terdiri dari berbagai sub network yang terintegrasi. Oleh karena itu, suatu IP Address harus bersifat unik untuk seluruh dunia. Tidak boleh ada satu IP Address yang sama dipakai oleh dua host yang berbeda.

IP Address terdiri dari bilangan biner sepanjang 32 bit yang dibagi atas 4 segmen. Tiap segmen terdiri atas 8 bit yang berarti memiliki nilai desimal dari 0 – 255. Range address yang bisa digunakan adalah dari 00000000.00000000.00000000.00000000 sampai dengan 11111111.11111111.11111111.11111111. Jadi, ada sebanyak 232 kombinasi address yang bisa dipakai diseluruh dunia (walaupun pada kenyataannya ada sejumlah IP Address yang digunakan untuk keperluan khusus). Jadi, jaringan TCP/IP dengan 32 bit address ini mampu menampung sebanyak 232 atau lebih dari 4 milyar host. Untuk memudahkan pembacaan dan penulisan, IP Address biasanya direpresentasikan dalam bilangan desimal. Jadi,

range address di atas dapat diubah menjadi address 0.0.0.0 sampai address 255.255.255.255.

IP Address dapat dipisahkan menjadi 2 bagian, yakni bagian network (bit-bit network/network bit) dan bagian host (bit-bit host/host bit). Bit network berperan dalam identifikasi suatu network dari network yang lain, sedangkan bit host berperan dalam identifikasi host dalam suatu network. Jadi, seluruh host yang tersambung dalam jaringan yang sama memiliki bit network yang sama. Sebagian dari bit-bit bagian awal dari IP Address merupakan network bit/network number, sedangkan sisanya untuk host. Garis pemisah antara bagian network dan host tidak tetap, bergantung kepada kelas network. Ada 3 kelas address yang utama dalam TCP/IP, yakni kelas A, kelas B dan kelas C. Perangkat lunak Internet Protocol menentukan pembagian jenis kelas ini dengan menguji beberapa bit pertama dari IP Address

Selain address yang dipergunakan untuk pengenalan host, ada beberapa jenis address yang digunakan untuk keperluan khusus dan tidak boleh digunakan untuk pengenalan host.

Address tersebut adalah :

#### 1. Network Address

Address ini digunakan untuk mengenali suatu network pada jaringan Internet. Misalkan untuk host dengan IP Address kelas B 167.205.9.35. Tanpa memakai subnet, network address dari host ini adalah 167.205.0.0. Address ini didapat dengan membuat seluruh bit host pada 2 segmen terakhir menjadi 0. Tujuannya adalah untuk menyederhanakan informasi routing pada Internet. Router cukup melihat network address (167.205) untuk menentukan kemana paket tersebut harus dikirimkan. Contoh untuk kelas C, network address untuk IP address 202.152.1.250 adalah 202.152.1.0. Analogi yang baik untuk menjelaskan fungsi network address ini adalah dalam pengolahan surat pada kantor pos. Petugas penyortir surat pada kantor pos cukup melihat kota tujuan pada alamat surat (tidak perlu membaca seluruh alamat) untuk menentukan jalur mana yang harus ditempuh surat tersebut. Pekerjaan “routing” surat-surat menjadi lebih cepat. Demikian juga halnya dengan router di Internet pada saat melakukan routing atas paket-paket data.

## 2. Broadcast Address

Address ini digunakan untuk mengirim/menerima informasi yang harus diketahui oleh seluruh host yang ada pada suatu network. Seperti diketahui, setiap paket IP memiliki header alamat tujuan berupa IP Address dari host yang akan dituju oleh paket tersebut. Dengan adanya alamat ini, maka hanya host tujuan saja yang memproses paket tersebut, sedangkan host lain akan mengabaikannya. Bagaimana jika suatu host ingin mengirim paket kepada seluruh host yang ada pada networknya ? Tidak efisien jika ia harus membuat replikasi paket sebanyak jumlah host tujuan. Pemakaian bandwidth akan meningkat dan beban kerja host pengirim bertambah, padahal isi paket-paket tersebut sama. Oleh karena itu, dibuat konsep broadcast address. Host cukup mengirim ke alamat broadcast, maka seluruh host yang ada pada network akan menerima paket tersebut. Konsekuensinya, seluruh host pada network yang sama harus memiliki address broadcast yang sama dan address tersebut tidak boleh digunakan sebagai IP Address untuk host tertentu. Jadi, sebenarnya setiap host memiliki 2 address untuk menerima paket : pertama adalah IP Addressnya yang bersifat unik dan kedua adalah broadcast address pada network tempat host tersebut berada. Address broadcast diperoleh dengan membuat seluruh bit host pada IP Address menjadi 1. Jadi, untuk host dengan IP address 167.205.9.35 atau 167.205.240.2, broadcast addressnya adalah 167.205.255.255 (2 segmen terakhir dari IP Address tersebut dibuat berharga 11111111.11111111, sehingga secara desimal terbaca 255.255). Jenis informasi yang dibroadcast biasanya adalah informasi routing.

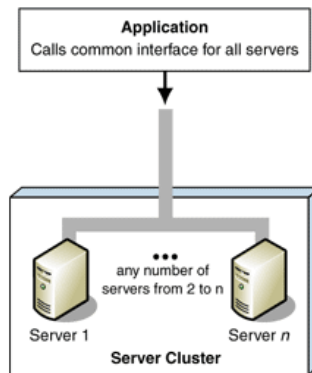
## 3. Netmask

Adalah address yang digunakan untuk melakukan masking / filter pada proses pembentukan routing supaya kita cukup memperhatikan beberapa bit saja dari total 32 bit IP Address. Artinya dengan menggunakan netmask tidak perlu kita memperhatikan seluruh (32 bit) IP address untuk menentukan routing, akan tetapi

cukup beberapa buah saja dari IP address yg kita perlu perhatikan untuk menentukan kemana packet tersebut dikirim.

Kaitan antara host address, network address, broadcast address & network mask sangat erat sekali – semua dapat dihitung dengan mudah jika kita cukup paham mengenai bilangan Biner. Jika kita ingin secara serius mengoperasikan sebuah jaringan komputer menggunakan teknologi TCP/IP & Internet, adalah mutlak bagi kita untuk menguasai konsep IP address tersebut. Konsep IP address sangat penting artinya bagi routing jaringan Internet.

### 3.3 Clustering Server



*Gambar 3.1 Skema Clustering Server*

Sumber: <https://i-msdn.sec.s-msft.com/dynimg/IC24476.gif>

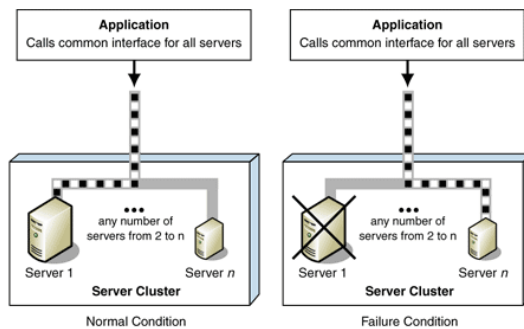
Server merupakan induk dari segala komputer yang terhubung pada sebuah jaringan yang berfungsi sebagai pengatur sistem jaringan. Kegagalan devices pada sebuah server bukan suatu yang tidak mungkin terjadi sehingga diperlukan solusi agar sistem jaringan tidak terganggu. (N.V. Patil, 2014). Asumsikan bila sebuah web server mati yang disebabkan oleh suatu hal (power supply mati atau yang lainnya), maka pengguna internet tidak akan bisa mengakses situs pada web server tersebut. Clustering menawarkan solusi untuk menangani perpindahan tugas atau pemerataan beban

dari satu server ke server yang lainnya apa bila terjadi kerusakan pada salah satu server.

Dalam dunia komputer yang dimaksud dengan Server Clustering adalah menggunakan lebih dari satu server yang menyediakan redundant interconnections, sehingga user hanya mengetahui ada satu sistem server yang tersedia dan komputer client tidak menyadari jika terjadi kegagalan pada sistem server karena tersedianya server sebagai redundant atau backup.(Muchtar,2012). Clustering Server dapat digunakan untuk Load Balancing cluster ataupun Failover clustering (Server HA).

### 3.3.1 Failover Clustering

Failover clustering menyediakan solusi high availability server dimana jika terjadi kegagalan pada perangkat keras seperti power supply mati yang menyebabkan server mati total maka server lain anggota cluster yang akan mengambil alih fungsi dari server yang mati, sehingga komputer client tidak mengetahui jika terjadi kegagalan pada server, karena proses yang dilakukan pada server yang gagal atau mati akan dilanjutkan oleh server cadangan.(Alan Hirt, 2009).



**Gambar 3.2 Contoh Failover Clustering**

Sumber : <https://i-msdn.sec.s-msft.com/dynimg/IC100847.gif>

Konsep konfigurasi failover cluster adalah membuat satu server sebagai master server dan server yang lain menjadi slave server dimana saat server dalam keadaan normal master server menangani semua request dari client.(Fikri Hidayat, 2012). Slave

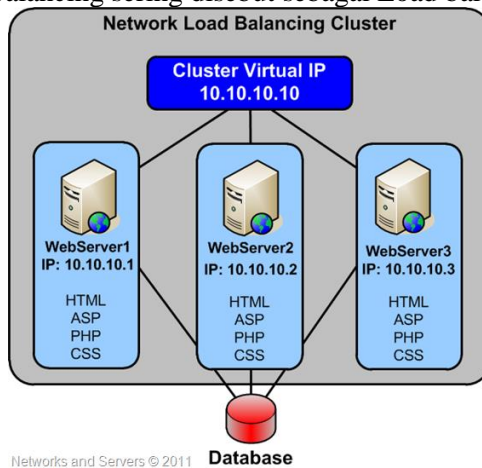


server akan mengambil alih tugas master server apabila master server tidak berfungsi atau mati. Failover server memiliki dua mode yaitu mode aktif-pasif (master-slave) dan (aktif-aktif). Aktif-pasif (master-slave) : Dua server atau lebih, yang melayani service jaringan hanya satu server saja, yang lain hanya sebagai cadangan jika terjadi kegagalan pada server aktif (master). Aktif-aktif (master-master) : Dua server yang kedua duanya bisa melayani jaringan dan saling mem-backup, jika salah satu server mati maka server yang lain akan menggantikannya. Kedua server ini memiliki data yang sama persis.

Biasanya failover menggunakan *shared storage* yang akan di gunakan bersamaan oleh lebih dari sebuah *server*. Tetapi ada juga yang tanpa menggunakan shared storage yaitu dengan menggunakan *mirroring hard disk server*.

### 3.3.2 Load Balancing Clustering

Load balancing cluster merupakan cluster server dimana anggota cluster server dikonfigurasi untuk saling berbagi beban yang berfungsi mendistribusikan request dari client ke anggota server Load balanced Cluster. Tipe konfigurasi Loadbalancing Cluster sering disebut Load balanced cluster, sedangkan teknologi platform Load balancing sering disebut sebagai Load balancers.



**Gambar 3.3 Contoh Load Balancing Cluster**

Sumber : <https://sint.cf/5XY7>

Secara umum cara kerja Load balancer adalah menerima incoming request dari client dan meneruskan request tersebut pada server tertentu jika dibutuhkan. Load balancer menggunakan beberapa algoritma yang berbeda untuk melakukan control traffic network. Tujuan algoritma Load balancer adalah untuk mendistribusikan beban secara pintar atau memaksimalkan kerja anggota server cluster.

### **3.3.3 Hubungan failover dan Loadbalancing**

Failover cluster dan Load Balancing memiliki manfaat dalam memanage server secara bersamaan. Tapi untuk failover cluster memiliki beberapa kekurangan dibanding Load Balancing yaitu : Peng-integrasi-an masing-masing server yang agak rumit, khususnya pada perangkat lunak yang digunakan harus memiliki setting yang sama antar server anggota cluster. Selain itu failover cluster terbatas untuk beberapa protocol seperti HTTP, samba dll.

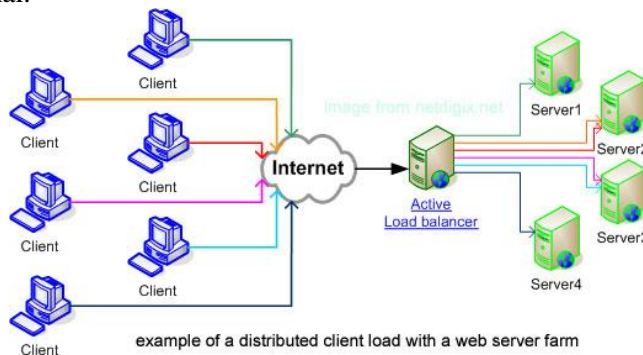
Sedang SLB (server load balancing) merupakan sebuah platform dan OS neutral. SLB dapat menyeimbangkan beban (Load) antar masing masing server. SLB juga mendukung beberapa network protocol dari HTTP hingga NFS, TCP dan UDP protocol. SLB di-desain secara simple sehingga tidak memerlukan interaksi antar server, sedikit melakukan trouble-shoot.

Dalam peng-konfigurasi-an server load balancing, load balancer diletakan didepan anggota server cluster dimana dianalogikan seperti sebuah gateway dari semua anggota server. Sedang failover clustering harus menggunakan protocol perangkat lunak yang sama untuk setiap server.

## **3.4 Load Balancing**

Load balancing adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan throughput, memperkecil waktu tanggap dan menghindari overload pada salah satu jalur koneksi. Load balancing digunakan pada saat sebuah server telah memiliki jumlah user yang telah melebihi maksimal kapasitasnya. Load balancing juga mendistribusikan beban kerja secara merata di

dua atau lebih komputer, link jaringan, CPU, hard drive, atau sumber daya lainnya, untuk mendapatkan pemanfaatan sumber daya yang optimal.



**Gambar 3.4 Contoh Penerapan Load Balancing**

Layanan Load Balancing memungkinkan pengaksesan sumber daya dalam jaringan didistribusikan ke beberapa host lainnya agar tidak terpusat sehingga unjuk kerja jaringan komputer secara keseluruhan bisa stabil. Ketika sebuah server sedang diakses oleh para pengguna, maka sebenarnya server tersebut sebenarnya sedang terbebani karena harus melakukan proses permintaan kepada para penggunanya. Jika penggunanya banyak maka prosesnya pun banyak. Session-session komunikasi dibuka oleh server tersebut untuk memungkinkan para pengguna menerima servis dari server tersebut. Jika satu server saja terbebani, tentu server tersebut tidak bisa banyak melayani para penggunanya karena kemampuan melakukan processing ada batasnya. Solusi yang paling ideal adalah dengan membagi-bagi beban yang datang ke beberapa server. Jadi yang melayani pengguna tidak hanya terpusat pada satu perangkat saja. Teknik ini disebut Teknik Load Balancing.

Adapun manfaat dari Load Balancing :

1. Menjamin Reliabilitas layanan berarti kepercayaan terhadap sebuah sistem untuk dapat terus melayani pengguna dengan sebaik-baiknya. Jaminan realibilitas memungkinkan pengguna dapat melakukan pekerjaan sebaik-baiknya dengan lancar melalui layanan tersebut.

- 2 Skalabilitas dan ketersediaan Jika dalam sebuah jaringan komputer jika hanya terdapat satu buah server mempunyai pengertian terdapat satu titik masalah. Seandainya tiba-tiba server itu mati maka layanan terhadap pengguna akan terganggu. Dengan melakukan penambahan server dan membentuk server farm maka skalabilitas akan meningkat dan selain itu faktor ketersediaan juga akan meningkat.

### 3.4.1 Metode Load Balancing

Berdasarkan perangkat yang digunakan, load balancing dibagi menjadi 3 jenis yaitu *load balancing* dengan hardware, load balancing dengan software dan load balancing kombinasi.()

#### 1. Load Balancing dengan Hardware / Switch

Sistem Load Balancing jenis ini diciptakan dengan menggunakan bantuan sebuah chip khusus yang sering disebut ASICs. ASICs berwujud sebuah microprocessor khusus yang hanya memproses algoritma dan perhitungan spesifik sehingga performa Load Balancing cukup handal karena hanya perhitungan dan logika Load Balancing saja yang dioptimasi didalamnya. Load Balancing jenis ini umumnya berwujud sebuah switch. Kelemahannya karena interfacenya yang kurang user friendly dan tingkat fleksibilitas perangkat juga rendah karena sebagian besar inteligennya sudah tertanam didalam hardware.

#### 2. Load Balancing dengan Software

Keuntungan yang paling menonjol menggunakan metode ini adalah : tingkat kemudahan pemakaian yang lebih user friendly. Keuntungan lain jika ada penambahan fitur atau fasilitas tambahan tidak perlu mengganti keseluruhan perangkat load balancing. Performa proses load balancing dipengaruhi oleh perangkat komputer yang digunakan, tidak bisa hanya mengandalkan kemampuan software yang canggih saja. Perangkat keras yang dapat mempengaruhi performa metode ini adalah kartu jaringan yang digunakan, besarnya RAM pada perangkat, media penyimpanan yang besar dan cepat, dsb. Sehingga performa metode ini sulit untuk bisa diperkirakan.

### 3 Load Balancing Kombinasi

Hardware yang dioptimasi dan diisi dengan platform berbasis Linux atau BSD yang dioptimisasi adalah konfigurasi yang biasanya digunakan untuk menjalankan software utama load balancing. Fleksibilitas yang luar biasa didapatkan mulai dari menggunakan hardware yang selalu up to date sampai dengan menggunakan operating system dengan patch terbaru. Sehingga waktu guna dari perangkat ini dapat lebih panjang daripada sebuah switch khusus yang tidak fleksibel. Solusi ini tentunya jauh lebih murah dibandingkan dengan solusi hardware khusus atau solusi software saja.

#### 3.4.2 Algoritma Load Balancing

Berikut algoritma dalam load balancing :

1. Round Robin

Algoritma round-robin mendistribusikan beban kepada semua server anggota cluster sehingga masing masing server mendapat beban yang sama dalam waktu yang sama. Round-robin cocok saat server anggota cluster memiliki kemampuan processing yang sama, jika tidak, beberapa server bisa jadi menerima request lebih dari kemampuan processing server itu sendiri sedang yang lainnya hanya mendapat beban lebih sedikit dari resource yang dimiliki.

2. Weighted round-robin.

Algoritma weighted round-robin melakukan perhitungan perbedaan kemampuan processing dari masing masing server anggota cluster. Administrator memasukkan secara manual parameter beban yang akan ditangani oleh masing masing server anggota cluster, kemudian scheduling sequence secara otomatis dilakukan berdasarkan beban server. Request kemudian diarahkan ke server yang berbeda sesuai dengan round-robin scheduling sequence.

3. Least-connection.

Algoritma Least-connection melakukan pengiriman request pada server anggota cluster, berdasarkan pada server mana yang memiliki fewest connections (koneksi paling sedikit).

4. Load-based.

Algoritma Load-based mengirimkan paket request ke server anggota cluster berdasarkan server mana yang memiliki beban terkecil.

## **BAB IV**

### **PELAKSANAAN PKL**

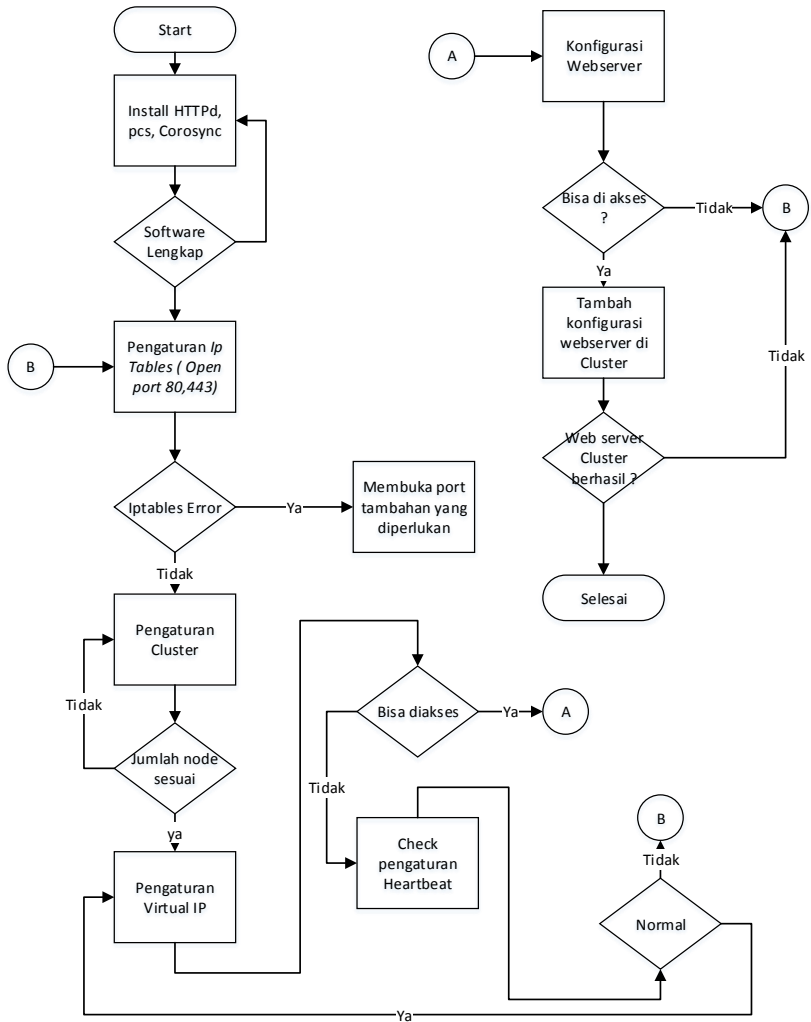
#### **4.1 Gagasan Multinode Server**

Sintask merupakan sebuah badan usaha yang bergerak pada bidang sosial media dimana dalam prakteknya dituntut agar memiliki infrastruktur yang mendukung. Salah satu infrastruktur utama yang diperlukan adalah infrastruktur jaringan yang dituntut untuk selalu siap sedia melayani permintaan dari penggunanya. jika saja infrastruktur jaringan ini rentan terhadap gangguan yang dapat terjadi sewaktu-waktu maka dapat dipastikan mengganggu kenyamanan pengguna dalam menggunakan jasa layanan sintask.

Gangguan infrastruktur jaringan yang dapat terjadi diantaranya adalah gangguan pada server yang menyebabkan gagalnya server untuk memenuhi permintaan pengguna. Hal ini merupakan gangguan fatal yang dapat menyebabkan terjadinya downtime. pada umumnya *downtime* server jarang terjadi pada server media sosial dikarenakan penyedia layanan berbasis media sosial itu sendiri telah melakukan antisipasi untuk mengurangi terjadinya downtime ini. Selain melakukan antisipasi pada server, penyedia layanan media sosial juga harus menyediakan infrastruktur yang kuat.

Gagasan untuk membuat infrastruktur jaringan yang kuat ini bertujuan guna melakukan antisipasi terhadap serangan dari pihak yang tidak bertanggung jawab. Selain itu munculnya ide untuk melakukan implementasi dengan menempatkan banyak server memperkuat stabilitas server itu sendiri. Dengan menempatkan banyak server yang dihubungkan dalam sebuah protocol tertentu yang disebut dengan multi node sever. Multimode server memungkinkan sistem tetap berjalan ketika terjadi kendala pada salah satu node dengan node lainnya sebagai backup. Dalam multinode yang diterapkan pada server sintask.com mengarah kepada node yang memiliki data yang identik.

## 4.2 Flowchart Implementasi Cluster Server



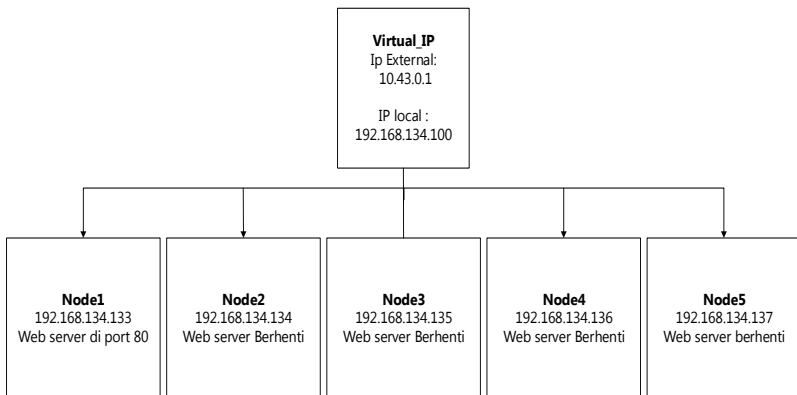
**Gambar 4.1 Flowchart Implementasi Cluster Server**



### 4.3 High Availability Cluster

Ada banyak scenario dan jenis cluster yang berbeda akan tetapi pada studi kasus yang penulis tangani berfokus pada *clustering* yang terbilang sederhana, terdapat 5 buah node yang melayani sebuah *web service*. Dalam kasus ini, penulis berfokus kepada ketersediaan apabila terjadi kegagalan pada *server* bukan untuk menyeimbangkan beban dalam jaringan ataupun meningkatkan kinerja. Tentu saja kasus ini dapat diperluas atau disesuaikan dengan apapun kebutuhan dengan kasus yang berbeda pula.

Tiap node yang ada pada cluster terdapat web server yang berjalan secara *redundant*. Agar node yang dibuat dapat melayani sebuah *web service* yang merupakan inti dari permasalahan, tentunya harus ada sebuah perantara yang menghubungkan antar node agar menjadi sebuah cluster yang khusus untuk melayani *web service*. Oleh karena itu *virtual ip* dibuat guna mewakili node cluster, terlepas dari berapa banyaknya node yang ada pada sebuah klien. klien hanya perlu mengetahui sebuah *virtual ip* dan atau alamat URL yang dituju, tidak perlu mencari alamat IP dari node atau alamat IP node yang aktif,



**Gambar 4.2 Skema Node Clustering Server**

Dari skema diatas terlihat bahwa terdapat sebuah IP yang mengatur node di dalam sebuah cluster. Apabila client mencoba

akses website Sintask, maka data yang berada pada node1 yang ditampilkan dikarenakan web server pada node1 aktif. Dalam kondisi diatas node yang lainnya dalam keadaan sleep state dimana node tersebut tidak melakukan apa-apa selain menunggu node1 gagal dalam melakukan tugasnya. Apabila pada suatu ketika terdeteksi bahwa node utama mengalami hal-hal menyebabkan node utama gagal dalam melakukan tugas nya maka node-node yang lainnya masih dapat mengambil alih guna menyediakan layanan yang sama kepada client layaknya node utama. Dari segi klien tidak akan pernah tahu bahwa terdapat masalah pada node1 dikarenakan informasi yang didapat tetaplah sama dikarenakan cara akses dan isinya pun identik. Ketika melakukan pemeliharaan pada infrastruktur jaringan terutama pada salah satu node, dapat dilakukan dengan melakukan pengalihan traffic antar node tanpa ada downtime

Untuk membangun sebuah cluster terdapat komponen dasar yang diperlukan, seperti layanan yang ingin diberikan kepada klien dan layanan ini harus selalu tersedia, klien punya hak akses penuh untuk mengakses layanan tersebut. Selanjutnya perlu adanya manajer sumber daya yang dapat mengatur resource yang ada pada sebuah cluster, seperti hal nya pacemaker. Ketiga, komponen yang dapat melakukan komunikasi antar node seperti Corosync. Yang terpenting yaitu konten manajer yang memungkinkan sinkronisasi file pada tiap node yang ada dalam cluster contohnya DRDB atau GlusterFS. Hal yang terakhir adalah sistem untuk manajemen kluster, pengaturan cluster pada tiap tiap node seperti PCS.

#### **4.4 Persiapan Implementasi *HACLUSTER***

Pada tahap ini konfigurasi tiap node disesuaikan seperti *hostname*, *static ip* tiap *node*, pengaturan firewall, dan instalasi komponen yang diperlukan.

##### **4.4.1 Pengaturan Hostname dan *Ip Address***

Pengaturan hostname disesuaikan dengan kebutuhan dan sesuai dengan urutan, dapat dirubah pada *etc/hosts*, pada studi kasus nama node diurutkan dari node1, node2, hingga node5 guna mempermudah dalam mengingat dan manajemen node. Cluster

yang dibuat dari kumpulan node ini menggunakan private IP static dengan address 192.168.134.0/24 dimulai dari IP 192.168.134.133 sampai dengan 192.168.134.137

#### 4.4.2 Konfigurasi Iptables

Pada pengaturan firewall beberapa port yang harus dibuka pada tiap node agar komunikasi data antar node yang dibutuhkan dapat dilakukan. Port yang dibutuhkan yaitu port UDP 5404 untuk Corosync, port TCP 2224 untuk PCS, port IGMP, port TCP 80 dan 443 untuk HTTP dan yang terakhir port multicast.

```
[root@node2 1]# sudo systemctl start pcsd
[root@node2 1]# sudo iptables -I INPUT -m state --state NEW -p udp -m multiport --dports 5404,5405 -j ACCEPT
[root@node2 1]# sudo iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 2224 -j ACCEPT
bash: udo: command not found...
[root@node2 1]# udo iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 2224 -j ACCEPT
bash: udo: command not found...
[root@node2 1]# sudo iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 2224 -j ACCEPT
[root@node2 1]# sudo iptables -I INPUT -p igmp -j ACCEPT
[root@node2 1]# sudo iptables -I INPUT -m addrtype --dst-type MULTICAST -j ACCEPT
[root@node2 1]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[ OK ]
[root@node2 1]#
```

**Gambar 4.3 Pengaturan Iptables**

Port yang telah dibuka pada iptables dapat dijadikan jalur transmisi data. Selain port yang dibutuhkan, port yang lainnya ditutup hal ini merupakan langkah antisipasi guna meminimalisir serangan yang dilakukan oleh orang yang tidak bertanggung jawab

#### 4.4.3 Instalasi Komponen yang Dibutuhkan

Untuk melakukan Implementasi cluster ini tentunya membutuhkan *software* pihak ketiga. Adapun perangkat lunak yang dibutuhkan tiap node yaitu Corosync, PCS, Pacemaker dan Httpd.

```
[1@node1 ~]$ sudo yum install corosync pcs pacemaker Httpd
...
Complete!
[1@node1 ~]$
```

```
[1@node2 ~]$ sudo yum install corosync pcs pacemaker Httpd
...
Complete!
[1@node2 ~]$
```

*Gambar 4.4 Proses Instalasi Komponen*

Setelah berhasil melakukan instalasi perangkat lunak yang dibutuhkan, maka dapat dilanjutkan ke tahap berikutnya.

## 4.5 Konfigurasi Cluster

Node node yang sudah memenuhi kriteria dapat melanjutkan ke tahap selanjutnya yaitu tahap konfigurasi nodes menjadi sebuah cluster. terdapat urutan saat melakukan konfigurasi, yaitu :

### 4.5.1 Menambahkan Password Default PCS

PCS diperlukan untuk mengelola node didalam cluster. hal ini memungkinkan untuk mempunyai antarmuka tunggal yang berfungsi guna mengelola semua node yang ada di dalam cluster. Sebelum dapat menggunakan PCS, ubah *password userme* “hacluster” dengan kata yang mudah di ingat pada setiap node.

```
[1@node1 ~]$ sudo passwd hacluster
Changing password for user hacluster.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[1@node2 ~]$
```

```
[1@node2 ~]$ sudo passwd hacluster
Changing password for user hacluster.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[1@node2 ~]$
```

*Gambar 4.5 Merubah Password Node*

Tahap ini diperlukan guna memberikan akses kepada orang yang berwenang, dengan demikian hanya orang yang diberi hak akses saja yang dapat melakukan konfigurasi. Setelah merubah

password, langkah selanjutnya adalah menjalankan PCS pada tiap tiap node dan lakukan autentikasi

#### 4.5.2 Autentikasi Node

Node yang telah dikonfigurasi dikelompokkan kedalam cluster yang diatur oleh PCSD,

```
[1@node1 ~]$ sudo systemctl start pcsd
[1@node1 ~]$ sudo pcs cluster auth node1 node2 node3 node4
node5
Username: hacluster
Password:
node01: Authorized
node02: Authorized
node03: Authorized
node04: Authorized
node05: Authorized
```

*Gambar 4.6 Pperintah Untuk Autentikasi Node*

Pada **Gambar 4.6** menegaskan jika node yang telah ter autentikasi. Dengan menjalankan PCSD, konfigurasi selanjutnya dapat dilakukan dari 1 node saja. Node yang ada di dalam yang cluster telah melalui tahap autentikasi tersinkronisasi dengan node yang lainnya. Perangkat lunak yang bertanggung jawab dalam melakukan sinkronisasi terhadap konfigurasi tiap node adalah corosync.

Denkrongan adanya sinkronisasi antar node ini berdampak pada efisiensi waktu yang dibutuhkan saat melakukan konfigurasi dikarenakan tidak diperlukan repetisi untuk memasukkan perintah yang sama pada tiap tiap node.

#### 4.5.3 Konfigurasi Utama Pada Cluster

Pada tahap ini, konfigurasi dilakukan untuk membentuk *nodes* menjadi sebuah cluster. pertama- tama lakukan *crm verify* untuk memastikan jika ada eror. Pada saat pengecekan pertama pasti akan ada sebuah notifikasi STONITH error. STONITH (Shoot The Other Node In The Head) merupakan mekanisme yang dirancang untuk memastikan bahwa tidak ada *nodes* yang saling rebutan

*Virtual IP*. STONITH di nonaktifkan karena *node* yang ada pada cluster berjalan secara parallel dan dapat hak akses penuh.

```
[1@node1 ~]$ sudo pcs property set stonith-enabled=false
[1@node1 ~]$
```

**Gambar 4.7 Perintah Untuk Mematikan STONITH**

Dengan merubah PCS *property* dikarena cluster yang dibuat merupakan cluster sederhana dengan empat buah *nodes*.

Langkah selanjutnya yaitu melakukan konfigurasi pada *quorum*. *Quorum* merupakan mekanisme yang mendefinisikan banyaknya minimal nodes yang ada didalam *cluster*. *Quorum* berguna ketika banyak nodes yang melayani komputasi skala besar. Ketika jumlah *nodes* yang tersedia sedikit maka quorum bertugas untuk menghentikan proses tersebut.

#### 4.6 Konfigurasi Virtual Ip

Setelah melakukan konfigurasi pada cluster, tentunya cluster tersebut akan ditugaskan sesuai dengan tujuannya yaitu melayani webserver, pertama cluster tersebut harus dapat menjalankan virtual ip sebagai jalur akses web service yang ada pada server sintask.com.

```
[1@node1 ~]$ sudo pcs resource create virtual_ip
ocf:hea rtbeat:IPaddr2ip=192.168.134.100
cidr_netmask=32 op mon itor interval=30s
[1@node1~]$ sudo pcs status resources
virtual_ip (ocf::heartbeat:IPaddr2): Started
[1@node1 ~]$
```

**Gambar 4.8 Perintah Untuk Membuat Resource Virtual Ip**

#### 4.7 Konfigurasi Webserver

Konfigurasi ini dilakukan agar web server dapat berjalan pada Virtual Ip yang telah dibuat pada langkah sebelumnya. Konfigurasi httpd agar dapat berjalan pada resource virtual ip dilakukan dengan merubah statemen listen pada file yang terletak /etc/httpd/conf/httpd.conf pada setiap node. Dengan kata lain merubah jalur komunikasi data dalam web server dan mengarahkannya ke *Virtual Ip* , Listen 192.168.134.100:80.

```
[l@node1~]$ echo "Listen 192.168.134.100:80"|sudo tee -append
/etc/httpd/conf/httpd.conf
[l@node1 ~]$
```

**Gambar 4.9 Sintaks Memindahkan Listen Httpd**

Setelah melakukan routing terhadap jalur yang dilalui webserver ke virtual IP, sekarang buat sebuah resource yang merupakan webserver pada PCS.

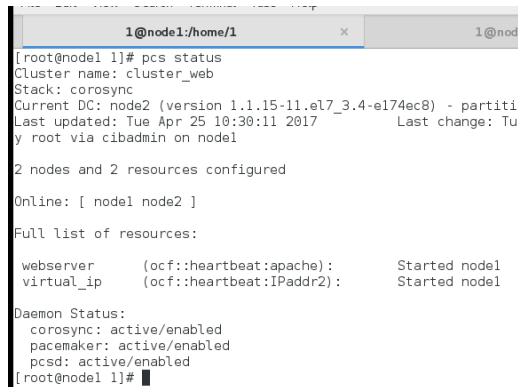
```
[l@node1~]$ sudo pcs resource create webserver ocf:heart
beat:apache          configfile=/etc/httpd/conf/httpd.conf
statusurl=" http://localhost/server-status" op monitor
interval=1min
[l@node1 ~]$
```

**Gambar 4.10 Perintah Membuat Resource Webserver**

Pembuatan resource ini dimaksudkan agar PCS dapat memanajemen webserver tersebut. Pada konfigurasi defaultnya, cluster akan mencoba menyeimbangkan resource yang ada pada cluster. Saat pertama kali menjalankan web server terdapat kendala yaitu node yang bukan pemilik *Virtual Ip* tidak bisa menjalankan web server karena httpd hanya menerima transaksi pada virtual ip saja. Oleh karena itu dibuat sebuah aturan bahwa web service dapat memiliki *Virtual Ip* secara bersama. Dengan menggunakan constraint berupa colocation. Kemudian untuk mencegah agar web server selalu berjalan setelah *virtual Ip* tambahkan constraint pada PCS berupa order dengan urutan *virtual Ip* diatas webserver.

#### **4.8 Error Checking**

Dalam sebuah implementasi *hacluster*, proses error checking dibutuhkan untuk memastikan tiap komponen apabila terjadi error pada saat tahap-tahap implementasi. *Error checking* dapat dilakukan dengan check status pada pcs. Jika terjadi error pada cluster maka status pada cluster akan menunjukkan error code dan status errornya.



```

1@node1:/home/1
[root@node1 ~]# pcs status
Cluster name: cluster_web
Stack: corosync
Current DC: node2 (version 1.1.15-11.el7_3.4-e174ec8) - partitioned
Last updated: Tue Apr 25 10:30:11 2017 Last change: Tue Apr 25 10:30:11 2017
y root via cibadmin on node1

2 nodes and 2 resources configured

Online: [ node1 node2 ]

Full list of resources:

webserver      (ocf::heartbeat:apache):      Started node1
virtual_ip     (ocf::heartbeat:IPaddr2):     Started node1

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
[root@node1 ~]#

```

**Gambar 4.11 Hasil Error Checking**

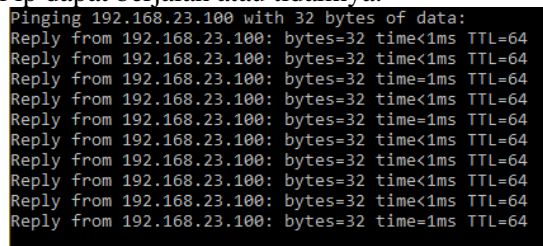
Berdasarkan hasil eror checking yang dipaparkan pada **Gambar 4.10** terlihat bahwa tiap elemen yang menyusun cluster telah berhasil berjalan tanpa ada Kendala. Terlihat dari resource pcs *virtual ip* dan *webserver* berjalan pada node1.

## 4.9 Pengujian sistem

Sebelum di implementasikan pada server utama sintask, cluster yang telah dibuat diuji terlebih dahulu agar memenuhi beberapa kriteria diantaranya yaitu skalabilitas, efisiensi sumber daya dan latency pada media penyimpanannya.

### 4.9.1 Pengujian Virtual Ip

Pada tahap ini, *virtual ip* diujikan dengan melakukan ping terhadap virtual ip tersebut. Dengan melakukan ping dapat diketahui jika virtual ip dapat berjalan atau tidaknya.



```

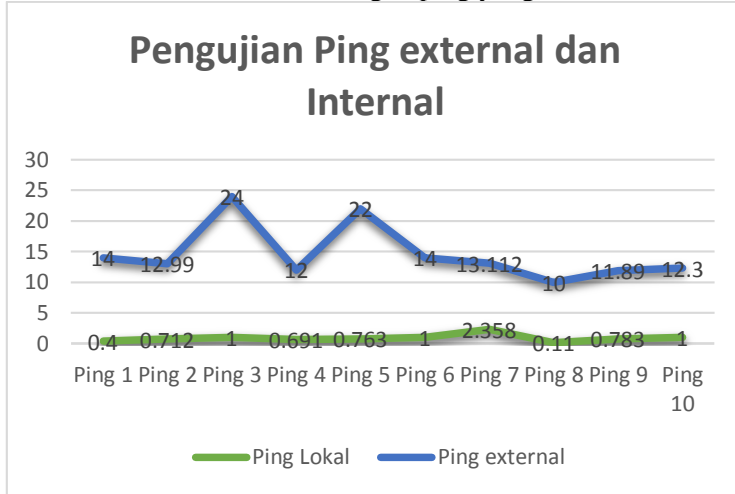
Pinging 192.168.23.100 with 32 bytes of data:
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64
Reply from 192.168.23.100: bytes=32 time<1ms TTL=64

```

**Gambar 4.12 Pengujian Virtual Ip**



Pada **Gambar 4.11** terlihat bahwa *virtual Ip* berhasil berjalan pada cluster. Hal ini dibuktikan dengan ping yang berhasil dilakukan.

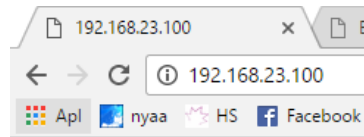


**Gambar 4.13** Grafik Pengujian Ping Pada Cluster

Pada **Gambar 4.13** pengujian ping pada cluster dilakukan dalam koneksi LAN dan koneksi external menggunakan Apache Jmeter. Apache Jmeter dapat mensimulasikan jumlah *client* yang melakukan ping. Dalam pengujian ping ini, server dibebani oleh 1000 *client* virtual yang melakukan ping terhadap server. Ping internal server dilakukan dengan melakukan remote server terhadap server cluster sehingga dapat menghasilkan nilai yang dan 0, dan ping external dilakukan dengan melakukan ping dari luar cluster. Latency yang diperoleh dari ping external jauh lebih besar dari ping internal hal ini dikarenakan ping external menggunakan jaringan internet.

#### 4.9.2 Pengujian Web Server

Pengujian pada web server ini dilakukan untuk mengecek apakah web server yang berjalan pada node berhasil melakukan tugasnya.



## NODE1

*Gambar 4.14 Pengujian Webserver*

Pada pengujian ini, web server dapat berjalan dengan baik. Terlihat pada **Gambar 4.13** web page yang ditampilkan merupakan webpage yang ada pada Httpd di dalam node1

### 4.9.3 Pengujian Failover Cluster

Pada pengujian ini, Cluster diuji dengan scenario mematikan salah satu server yang ada di *cluster*.

```
[root@node1 1]# sudo pcs cluster stop node1
node1: Stopping Cluster (pacemaker)...
node1: Stopping Cluster (corosync)...
[root@node1 1]# █
```

*Gambar 4.15 Node1 dinonaktifkan*

Node1 merupakan node aktif, yang menjalankan *virtual ip* dan webserver. Skenario mematikan salah satu node aktif dapat menjadi simulasi apabila node utama mengalami gangguan yang menyebabkan gagalnya node untuk melayani permintaan dari user.

Jika sistem yang diterapkan tidak menggunakan multi node clustering maka apabila server mati, layanan yang bergantung pada serverpun tidak dapat di akses.



## NODE2

*Gambar 4.16 Webserver Node2*

Pada **Gambar 4.16** web page yang tampil pada browser user merupakan web page yang ada pada node2. Hal ini menandakan sistem dapat berjalan dengan baik.

```
[root@node2 1]# pcs status
Cluster name: cluster_web
Stack: corosync
Current DC: node2 (version 1.1.15-11.el7_3.4-e174ec8) - partition with quorum
Last updated: Tue Apr 25 13:14:36 2017      Last change: Tue Apr 25 09:52:25 2017 b
y root via cibadmin on node1

2 nodes and 2 resources configured

Online: [ node2 ]
Offline: [ node1 ]

Full list of resources:

webserver      (ocf::heartbeat:apache):      Started node2
virtual_ip     (ocf::heartbeat:IPaddr2):     Started node2

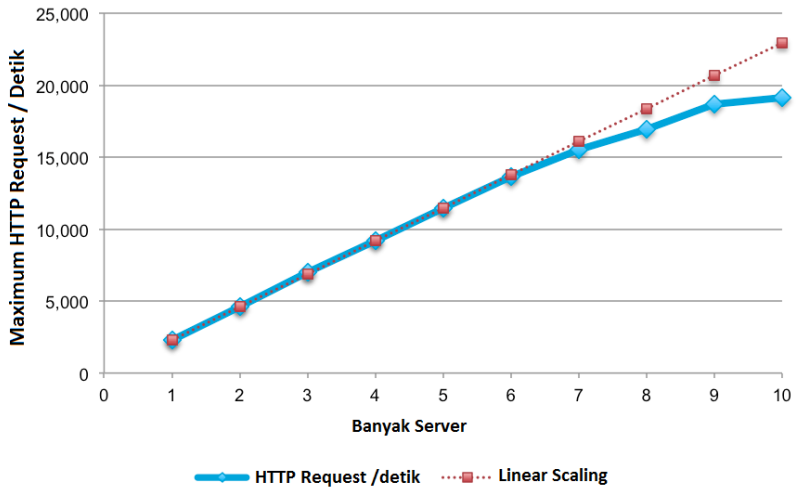
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@node2 1]#
```

**Gambar 4.17 Backend Server Clustering**

Perpindahan node yang dilakukan oleh sistem dapat dengan mudah mengalihkan lalu lintas data dari node1 ke node2. Hal ini dibuktikan oleh **Gambar 4.17**. sistem akan memindahkan jalur dari node1 ke node2 yang aktif, sehingga tidak terjadi downtime.

#### 4.9.4 Pengujian Efisiensi Skalabilitas

Efisiensi dalam sebuah cluster server sangatlah penting, karena jumlah resource yang terpakai oleh cluster tidaklah sedikit. Pengujian efisiensi pada cluster dilakukan agar cluster yang telah dibangun dapat mengalokasikan resource yang ada secara optimal. Resource yang dimaksud adalah, bandwidth jaringan, dan jumlah node. Pengujian ini menggunakan aplikasi pihak ketiga bernama wireshark untuk memonitor jaringan dan Apache JMeter yang berguna pada saat menguji cluster terhadap parameter pengujian yang berupa latency, scalability performance, dan Maximum HTTP Request.



*Gambar 4.18 Skalabilitas Server*

Pada **Gambar 4.18** merupakan visualisasi hasil dari pengujian skalabilitas cluster. dengan detail sebagai berikut :

Banyak Jumlah Node	Request HTTP/ detik Ideal	Request HTTP / detik Keluaran	Index Skalabilitas Ideal	Keterangan
1	2653	2781	1.843	Ideal
2	4890	4900	1.478	Ideal
3	7230	7240	1.361	Ideal
4	9847	9849	1.213	Ideal
5	11950	11940	1.200	Ideal
6	14340	14171	1.160	Sedikit dibawah ideal
7	16641	15802	1.139	Dibawah ideal

8	18954	17031	1.127	Tidak ideal
9	21380	18320	1.100	Tidak ideal
10	23518	19870		Tidak ideal

**Tabel 4.1 hasil pengujian skalabilitas server**

Dari **Tabel 4.1** diatas, diperoleh hasil optimal dengan node berjumlah 5 buah, dengan maksimum HTTP request mencapai 11.940 request per detik. Pengujian dilakukan dengan menghitung banyak nya *request* yang dapat diolah oleh cluster dengan satuan request per detik, kemudian dibandingkan dengan request HTTP/detik yang ideal. Banyaknya *request* HTTP ideal didapatkan dari melakukan simulasi dengan spesifikasi server dummy yang menyerupai dengan server sintask. Server dummy pada pengujian ini menggunakan jaringan LAN dimana cluster berada pada sebuah subnet yang sama.

*Node* yang ada pada *cluster* dapat melebihi dari 5 buah namun hal ini tidaklah efisien. Pada saat jumlah node melebihi 5 buah terjadi penurunan antara batas maksimal request HTTP dengan skalabilitas server. Penurunan rasio ini dapat disebabkan oleh beberapa faktor seperti *bottleneck* pada jaringan cluster sehingga mengurangi efisiensi *cluster* secara menyeluruh. *Bottleneck* pada jaringan *internal cluster* dapat disebabkan oleh lalu lintas data yang ada pada cluster membanjiri infrastruktur jaringan yang menyusunnya. Semakin banyak node pada cluster, semakin banyak data yang harus ditransmisikan agar *content* pada node menjadi identik.

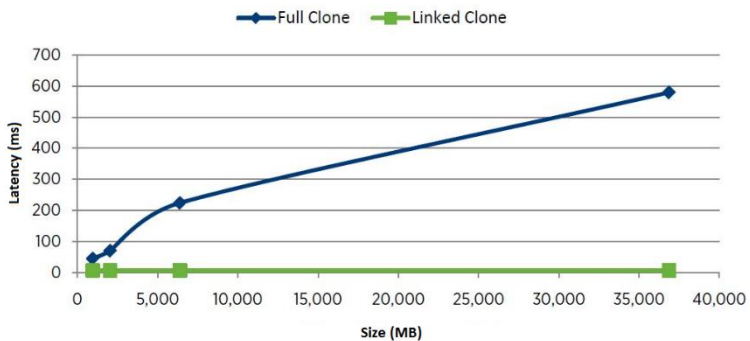
Efisiensi skalabilitas dapat berubah mengikuti komponen yang menyusun *cluster*. Contoh ketika *upgrade* prosessor, adanya penambahan ram, dan penambahan bandwidth internet, maka efisiensinya pun ikut berubah. Ketika terdapat perubahan yang terjadi pada komponen penyusun *cluster* maka pengujian dilakukan kembali ini agar data yang diperoleh akurat.

#### **4.9.5 Pengujian Latency Storage**

Pada pengujian ini, penyimpanan data yang ada pada cluster diuji agar mengetahui penempatan media penyimpanan yang ideal

pada cluster. Pada pengujian ini, metode Linked Clone dan metode Full Clone diuji. Parameter pengujian adalah latency yang menandakan waktu yang dibutuhkan oleh storage untuk mengirim dan menerima data.

*Full clone* merupakan metode menyimpan data dimana data tersebut terdistribusi pada tiap-tiap node. Node aktif mendistribusikan data yang diterima kepada node pasif sehingga data yang ada pada setiap *node* identik. Sedangkan *Linked clone* merupakan metode yang menempatkan media penyimpanan terpisah dengan cluster kemudian tiap cluster hanya menyimpan *link block* data yang tersimpan pada media penyimpanan.



**Gambar 4.19 Perbedaan metode penyimpanan**

Pada **Gambar 4.19** terlihat bahwa media penyimpanan dengan metode linked clone memiliki latency yang rendah hal ini dikarenakan data yang dibutuhkan berada pada satu lokasi dan tiap node pada cluster merujuk pada block data yang disimpan pada sebuah storage node saja. Tentunya metode linked clone ini memiliki resiko apabila media penyimpanan yang ada pada cluster ini terjadi kegagalan tidak ada backup yang bertindak sebagai media penyimpanan.

Pada metode *full clone* media penyimpanan memiliki *backup* sejumlah node yang ada pada cluster sehingga apabila node gagal masih terdapat cadangan yang siap menggantikan. Disamping redundancy yang ditawarkan, terdapat kelemahan pada metode full clone yaitu akses time dengan *latency* yang tinggi dikarenakan

adanya sinkronisasi pada setiap transaksi data yang memastikan bahwa tiap node memiliki data yang identik. Proses sinkronisasi tersebut memakan bandwidth, sehingga proses transaksi data yang akan dilakukan menunggu terlebih dahulu hingga proses sinkronisasi selesai. Apabila media penyimpanan *node* yang terdapat pada cluster menggunakan jalur internet maka dapat dipastikan bahwa bandwidth yang dibutuhkan cukup tinggi mengingat sinkronisasi tiap media penyimpanan pada full clone melibatkan aliran data yang cukup banyak pada tiap transaksinya.

**Gambar 4.19** menunjukkan hasil *latency* dari *linked clone* yang mendekati 0 dikarenakan, lokasi media penyimpanan berada pada local disk dengan rata-rata *latency* dari tiap transaksi 0.1 hingga 1 ms yang bergantung pada load harddisk saat pengujian.

Besaran data yang digunakan sebagai parameter pengujian adalah 1024 MB (1 GB) dan 37000 MB (36.1 GB) adalah batas maksimal ruang kosong dari harddisk yang digunakan. Setelah partisi selesai dibuat, kemudian data di upload dari sisi client dengan menggunakan protokol SMB *filesharing*. Protokol SMB ini memungkinkan pertukaran data dari client dengan sistem operasi Windows 10 dengan mudah.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Adapun kesimpulan yang dapat diperoleh dari praktek kerja lapangan ini adalah sebagai berikut.

1. Implementasi High availability cluster yang dilakukan pada server sintask berhasil dilakukan dengan node berjumlah 2 buah pada dummy *server* dan 5 node pada *server* sintask.
2. Penerapan cluster server dapat mengurangi downtime yang terjadi apabila terdapat masalah. Dengan menggunakan clustering server, apabila node yang sedang menjalankan servicenya gagal, maka masih terdapat node yang lain
3. Dari hasil pengujian sistem diperoleh efisiensi skalabilitas optimal pada cluster berjumlah 5 buah node.
4. Metode penyimpanan yang sesuai pada cluster adalah metode linked clone, dikarenakan metode ini tidak membebani jaringan internal cluster sebagaimana pada saat metode full clone melakukan replikasi data yang ada pada tiap node nya

#### **5.2 Saran**

Berdasarkan pengalaman yang didapatkan ketika penulis melaksanakan Praktek Kerja Lapangan (PKL) dan menyelesaikan dokumentasi, saran yang dapat disampaikan diantaranya :

Untuk mengembangkan hasil dokumentasi, dibutuhkan ketelitian, tenaga, serta waktu yang lebih mengingat clustering server memerlukan node dengan jumlah server yang jauh lebih banyak.

Dalam penerapan clustering server yang dilakukan, dapat dikembangkan lebih lanjut lagi seperti penambahan HAproxy dan *loadbalancing* sebagai penyeimbang beban, dan DRDB (Distributed Replicated Block Device) sebagai manajemen konten yang mengatur dan memastikan agar isi dari web page yang ada pada setiap node identik.



## DAFTAR PUSTAKA

- Budi, Setiya. 2013. *Implementasi Cluster Pada Web Server Berbasis Cloud Computing. Skripsi*. Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Kalijaga. Yogyakarta.
- Calzolari, Federico; Arezzini, Silvia; Ciampa, Alberto eds. *High Availability Using Virtualization. IOP Sciense*, 2010.
- Don Frima, Iyoga.(2012). *Implementasi Sistem Multiple-Computer Cluster Menggunakan Linux Enterprise Real Application Cluster (LINUXERAC) berbasis Metode Storage Area Network (DRBD) serta Analisa High Performance dan High Availability*. Depok : Universitas Indonesia.
- Febriani, Tania Rizky. 2011. *Implementasi dan Analisa Sistem Failover Virtual Computer Server. Skripsi*. Fakultas Teknik Universitas Indonesia.
- Hidayat, Fikri. 2012. *Implementasi dan Analisa Redudansi dan High Availability Dalam Server Untuk Diskless Thin Client Berbasis Storage Area Network. Skripsi*. Fakultas Teknik Universitas Indonesia.
- Hirt, Alan. 2009. *Pro SQL Server 2008 Failover Clustering*, Apress, New York.
- Lukitasari, Desi & Oklilas, Ahmad Fali. 2010. *Analisis Perbandingan Load Balancing Web Server Tunggal Dengan Web server Cluster Menggunakan Linux Virtual Server*. Universitas Sriwijaya : Jurnal Generic Fakultas Ilmu Komputer.
- Muchtar dkk. *Implementasi Failover Clustering Pada Dua Platform Yang Berbeda Untuk Mengatasi Kegagalan Fungsi Server*. Fakultas Teknik Universitas Hasanuddin.
- Patil, N.V., et al. (2014). *Cost Effective Failover Clustering*. International Journal of Research in Engineering and Technology. Vol 3: hal 3.
- Pratama, I Putu Agus Eka. 2014. *Smart City Beserta Cloud Computing dan Teknologi-teknologi Pendukung Lainnya*. Bandung:Informatika.

- Rasian, R dan Mursanto, P. 2009. *Perbandingan Kinerja Pendekatan Virtualisasi*. Jurnal Sistem Informasi. Magister Teknik Informatika Univeristas Indonesia.
- Software Development for High Availability Cluster : [www.drbd.org](http://www.drbd.org). Oleh LINBIT HA- Solutions. Vienna, Austria. 2008-2011. diakses pada Bulan 3 April 2017.
- Sugianto, Masin Vavai. 2011. “Kelebihan dan Kekurangan Virtualisasi. (<https://www.excellent.co.id/product-services/vmware/keuntungan-teknologi-virtualisasi-cloud-computing> , diakses 5 April 2017).

# LAMPIRAN

## LAMPIRAN A

### FORM AKTIFITAS HARIAN PKL

#### AKTIVITAS HARIAN PKL

Nama : Almer Hafiz Wandakana  
 NIM : 130805011  
 Lokasi PKL : CV. Sintask  
 Waktu Pelaksanaan : 06 Februari 2017 - 31 Maret 2017

No.	Nama Penanggung Jawab/Jabatan	Pelaksanaan PKL			Keterangan
		Tanggal	Lokasi	Aktivitas	
1	Tutde Suputrawan / Network and Server Engineer	06-02-2017	Gedung Animasi, Balai Diklat Industri	Riset tentang bagaimana menyemunkikan dan melindungi port dari scan attager. Port tidak boleh mati dan bisa digunakan oleh developer Sintask.com. OS yang digunakan Centos 6.	
2	Aditya Wikardiyan/CEO	07-02-2017	Gedung Animasi Balai Diklat Industri	Implementasi Port Knocking dengan menggunakan Knockid sebagai penambahan keamanan pada Dummy (Laptop pribadi). hasil yang didapat pada saat dummy CentOS tidak kuat mengimangi 3 buah Virtual Machine (1 server 5gb ram, 2 buah client 2gb ram) sehingga cv	
3	Aditya Wikardiyan /CEO	08-02-2017	Gedung Animasi Balai Diklat Industri	Riset Penggunaan dan Implementasi Advanced Policy Firewall pada centos dan Melanjutkan Implementasi Port Knocking menggunakan Knockid pada server Sintask.com	

Komisi Praktek Kerja Lapangan PS. Teknik Informatika FMIPA Universitas Udayana

16

4	Tutde Suputrawan/ Network And Server Engineer	09-02-2017	Gedung Animasi, Balai Diklat Industri	Lanjutan Implementasi Knockid pada centos	
5	Tutde Suputrawan/ Network And Server Engineer	10-02-2017	Gedung Animasi, Balai Diklat Industri	Lanjutan Implementasi Knockid pada centos	
6	-	11-02-2017	-	Libur	
7	-	12-02-2017	-	Libur	
8	Tutde Suputrawan/Network and Server Engineer	13-02-2017	Gedung Animasi, Balai Diklat Industri	Research and APF dan Implementasi Mengamankan Port Pada Dummy	
9	Tutde Suputrawan/Network and Server Engineer	14-02-2017	Gedung Animasi, Balai Diklat Industri	Memperbaiki Komputer Sintask dan membuat rancangan disaster recovery plan	
10	-	15-02-2017	-	Libur Nasional	
11	Tutde Suputrawan/Network and Server Engineer	16-02-2017	Gedung Animasi, Balai Diklat Industri	Implementasi disaster recovery plan pada harddisk yang rusak	
12	Tutde Suputrawan/Network and Server Engineer	17-02-2017	Gedung Animasi, Balai Diklat Industri	research port seouring dengan metode fail2ban ( bruteforce prevention)	
13	-	18-02-2017	-	Libur akhir Pekan	
14	-	19-02-2017	-	Libur Akhir pekan	

Komisi Praktek Kerja Lapangan PS. Teknik Informatika FMIPA Universitas Udayana

26

15	Ahmad Hanafi/Business Director & Marketing strategy	20-02-2017	Gedung Animasi, Balai Diklat Industri	melanjutkan research APF dan port seouring	
16		21-02-2017			
17		22-02-2017			
18		23-02-2017			
19		24-02-2017			
20		25-02-2017			
21		26-02-2017			
22		27-02-2017			
23		28-02-2017			
24		01-03-2017			
25		02-03-2017			

26		03-03-2017			
27		04-03-2017			
28		05-03-2017			
29		06-03-2017			
30	Tutde Suputrawan/Network and Server Engineer	07-03-2017	Gedung Animasi, Balai Diklat Industri	Research clustering server menggunakan Openstack	
31	Tutde Suputrawan/Network and Server Engineer	08-03-2017	Gedung Animasi, Balai Diklat Industri	Research clustering server menggunakan Openstack, dan melakukan persiapan pameran industri kreatif	
32	Tutde Suputrawan/Network and Server Engineer	09-03-2017	Gedung Animasi, Balai Diklat Industri	Mengikuti pameran industri kreatif mewakili jurusan ilmu komputer universitas Udayana dan sintak	
33		10-03-2017			
34		11-03-2017			
35		12-03-2017			
36		13-03-2017			

37	Tutde Suputrawan/Network and Server Engineer	14-03-2017	Gedung Animasi, Balai Diklat Industri	research mengenai cluster di jaringan	
38		15-03-2017			
39		16-03-2017			
40		17-03-2017			
41		18-03-2017			
42		19-03-2017			
43		20-03-2017			
44		21-03-2017			
45		22-03-2017			
46		23-03-2017			
47		24-03-2017			


48		25-03-2017			
49		26-03-2017			
50		27-03-2017			
51		28-03-2017			
52		29-03-2017			
53	Tutde Suputrawan/Network and Server Engineer	30-03-2017	Gedung Animasi, Balai Diklat Industri	Implementasi Clustering Server, High Availability cluster server mencegah terjadinya downtime	
54	Tutde Suputrawan/Network and Server Engineer	31-03-2017	Gedung Animasi, Balai Diklat Industri	Finalisasi clustering server pada server Sintak	

.....  
Pembimbing Lapangan,

.....

## LAMPIRAN B

### SURAT KETERANGAN SELESAI MELAKSANAKAN PKL



**CV. SINTASK**  
Pengembangan Perangkat Lunak  
Jalan WR. Supratman No. 302, Tohpati, Denpasar  
+62 812-4697-0994  
career@sintask.com

---

Nomor : 002/ND/STK/IV/17  
Lampiran : -  
Perihal : Surat keterangan telah selesai PKL


Kepada Yth :  
Ketua Komisi Praktek Kerja Lapangan Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Udayana  
Ditempat,

Dengan Hormat,

Bersama surat ini, Mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Udayana yang namanya tercantum di bawah ini :

Nama	NIM
Aditya Caesar Bagaskara	1208605034
Almer Hafiz Wandalaksana	1308605011

Telah menyelesaikan kegiatan Praktek Kerja Lapangan di CV. SinTask dari tanggal 06 Februari 2017 sampai tanggal 31 Maret 2017. Selama melaksanakan kegiatan Praktek Kerja Lapangan, mahasiswa yang bersangkutan telah bekerja dengan baik.

Denpasar, 03 April 2017  
CV. SinTask, Direktur  
  
Aditya Wikardiyana

www.sintask.com