

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Operating System CentOS**

CentOS (Community ENTerprise Operating System) adalah sebuah distribusi linux sebagai bentuk dari usaha untuk menyediakan platform komputasi berkelas enterprise yang memiliki kompatibilitas kode biner sepenuhnya dengan kode sumber yang menjadi induknya, Red Hat Enterprise Linux (RHEL). RHEL merupakan distribusi linux berbayar yang menyediakan akses terhadap update atas perangkat lunak dan beragam jenis dukungan teknis. Distribusi linux ini sebenarnya merupakan gabungan dari sejumlah perangkat lunak yang didistribusikan dibawah lisensi perangkat lunak bebas dan kode sumber atas paket perangkat lunak ini dirilis ke public oleh Red Hat sebagai bagian dari kesepakatan dalam lisensi yang digunakan.

Para pengembang CentOS menggunakan kode sumber dari Red Hat, dikompilasi dengan tujuan membuat sebuah produk final yang sangat mirip dengan RHEL. Semua hal-hal yang berkaitan dengan merek dagang ataupun logo kemudian diubah disebabkan Red Hat tidak mengizinkan mereka untuk mendistribusikan ulang logo tersebut.

CentOS tersedia secara gratis, dukungan teknis utamanya disediakan terhadap para pengguna melalui milis, forum berbasis web, ataupun chat. Proyek CentOS tidak berafiliasi dengan Red Hat, sehingga proyek CentOS berjalan tanpa mendapatkan bantuan apapun dari Red Hat. Untuk penggalangan dana, CentOS berbasis donasi dari para pengguna serta sponsor dari perusahaan-perusahaan yang menggunakannya.

Jadi, keberadaan CentOS adalah tidak lain untuk menyediakan platform komputasi skala perusahaan yang bebas, dan berusaha untuk mempertahankan 100% kompatibilitas binernya terhadap sumber hulu (RHEL). Tak peduli apapun alasannya, sejatinya CentOS tidak lain adalah clone dari RHEL yang berdasarkan kaidah lisensi GPL adalah sah-sah saja. Red Hat dan produknya (RHEL) pada prinsipnya hanya dilindungi aturan merek dagang, dan ini cukup memberi ruang untuk Red Hat mengembangkan bisnis layanan yang sukses. (The CentOS Project, 2020)



(Sumber : <https://www.centos.org/>)

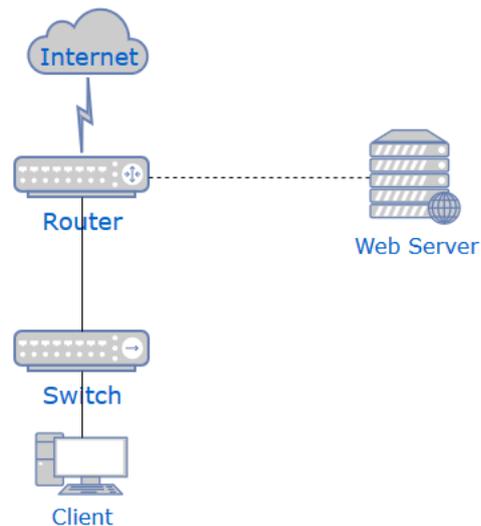
**Gambar 2.1 CentOS**

## **2.2 Web Server**

Web Server adalah sebuah software yang memberikan layanan berbasis data dengan menggunakan protocol HTTP atau HTTPS dari client menggunakan aplikasi web browser untuk request data dan server akan mengirim data dalam bentuk layanan web dan pada umumnya berbentuk dokumen HTML. Halaman web yang diminta bias terdiri dari berkas teks, video gambar file dan banyak lagi (Alam & Firmansyah : 2017).

Salah satu Program Web Server adalah Apache, Apache merupakan web sever 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 dengan penggunaannya. Aplikasi lain yang memiliki fungsi sebagai web server diantaranya apache Tomcat, Microsoft windows Server 2003 Internet Information Services (IIS). Lighttpd, Sun Java System Web Server, Xitami Web Server, dan

Zeus Web Server. Arsitektur request dan response web server dapat dilihat pada gambar 2.1 dibawah ini



Gambar 2.2 Arsitektur Web Server

### 2.3 Apache

Apache adalah sebuah nama web server yang bertanggung jawab pada request-response HTTP dan logging informasi secara detail (kegunaan dasarnya). Selain itu, Apache juga diartikan sebagai suatu web server yang kompak, modular, mengikuti standar protokol HTTP, dan tentu saja sangat digemari. Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah. Apache merupakan perangkat lunak sumber terbuka dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation.

Saat ini ada dua versi Apache yang bisa dipakai untuk server produksi yaitu versi mayor 2.0 dan versi mayor 1.3. Apache merupakan webserver yang paling banyak digunakan saat ini. Hal ini disebabkan oleh beberapa sebab, di

antaranya adalah karena sifatnya yang opensource dan mudahnya mengkostumisikannya. Di antaranya dengan menambahkan support secure protocol melalui ssl dan konektifitasnya dengan database server melalui bahasa scripting PHP. (Artanto : 2013)



(Sumber : <https://httpd.apache.org/>)

**Gambar 2.3** Apache

#### **2.4 Clustering Server**

Clustering server merupakan yang menggabungkan beberapa sumber daya yang bekerja bersama -sama sehingga tampak seolah -olah merupakan suatu sistem tunggal (Irfani, 2015:1). Dalam dunia TI sendiri Clustering server merupakan sebuah infrastruktur yang menggunakan lebih dari satu server yang menyediakan redundant interconnection, sehingga pengguna hanya mengetahui hanya ada satu sistem yang berjalan dan pengguna tidak akan menyadari jika terjadi kerusakan fisik server maupun kegagalan dari sistem server sendiri itu sehingga server mati karena tersedia server lain berguna sebagai backup

#### **2.5 Cluster Computing**

Komputer kluster adalah sekumpulan komputer (umumnya server jaringan) independen yang bekerja secara bersama sebagai sumber daya komputasi tunggal yang terintegrasi dan terlihat oleh klien seolah-olah komputer-komputer tersebut adalah satu buah unit komputer. Proses menghubungkan beberapa komputer agar dapat bekerja seperti itu dinamakan dengan clustering.

Komponen cluster biasanya saling terhubung melalui sebuah interkoneksi yang sangat cepat, atau bias juga melalui jaringan local (LAN)

Dalam menjalankan fungsinya Cluster Computing diklasifikasikan menjadi tiga, yaitu :

1. High-availability Cluster, yang juga disebut failover cluster pada umumnya diimplementasikan untuk tujuan meningkatkan ketersediaan layanan yang disediakan oleh kluster. Elemen kluster memiliki node-node redundan yang akan digunakan untuk menyediakan layanan ketika salah satu komponen mengalami kegagalan. Dibutuhkan dua buah node sebagai syarat minimum suatu kluster untuk dapat melakukan redundansi.

#### Macam-macam Failover

##### a. Active/passive Failover

Pada mode ini terdapat dua komponen, yaitu satu menjadi komponen atau node aktif dan yang lainnya pasif. Node aktif bertugas untuk melakukan eksekusi terhadap aplikasi atau tugas tertentu, sedangkan node pasif berstatus standby dengan tidak melakukan tugas apapun sampai mendeteksi bahwa terdapat masalah pada node utama/aktif. Pada saat node utama mengalami kegagalan, node pasif akan mengambil alih tugas yang tadinya dilakukan oleh node utama.

##### b. Active/active Failover

Pada mode ini, semua node berstatus aktif dengan menjalankan masing-masing aplikasi atau proses yang merupakan beban kerja masing-masing node. Apabila satu node aktif mengalami kegagalan, node aktif lain akan mengambil alih beban kerja node yang gagal tersebut sehingga node aktif yang masih berfungsi menjalankan seluruh aplikasi dan proses yang ada. Tujuan dari active/active failover ini adalah untuk mencapai *load balancing*. Yang membedakan mode ini dengan *load balancer cluster* yaitu adanya konfigurasi untuk redundancy diantara kedua node yang aktif.

2. Load-balancing Cluster, kluster kategori ini beroperasi dengan mendistribusikan beban kerja secara merata kepada beberapa node yang

bekerja dibelakang (back-end node) sehingga beban kerja disisi server menjadi lebih ringan. Tujuan dari load balancing adalah mempersingkat waktu rata-rata pengerjaan tugas pada server dan ketersediaan layanan yang tinggi.

3. High Performance Cluster, cluster yang bertujuan untuk meningkatkan kemampuan komputasi dengan memanfaatkan utilitas perangkat secara maksimal. (Rosalia, Rendi & Ratna : 2016)

## 2.6 Load Balancing

*Load Balancing* adalah teknik membagi jumlah pekerjaan yang harus dilakukan komputer antara dua lebih komputer sehingga setiap komputer melakukan pekerjaan dalam jumlah dan waktu yang sama serta secara umum semua pengguna mendapatkan akses cepat. *Load Balancing* digunakan pada saat sebuah server telah memiliki jumlah pengguna yang telah melebihi kapasitas maksimal dari server tersebut (Supramana & Prisma: 2016)

*Load Balancer* adalah perangkat lunak yang berguna untuk membagi beban agar menjadi seimbang dalam melayani suatu dari para user. Pembagian beban permintaan atau akses dari user akan merata ke semua server yang ada, kinerja server juga menjadi lebih baik.

*Load Balancer* menggunakan beberapa peralatan yang sama untuk menjalankan tugas yang sama. Hal ini memungkinkan pekerjaan dilakukan dengan lebih cepat dibandingkan apabila dikerjakan oleh hanya 1 peralatan saja dan dapat meringankan beban kerja peralatan, serta mempercepat waktu respons.

*Load Balance* bertindak sebagai penengah diantara layanan utama dan pengguna, dimana layanan utama merupakan sekumpulan server/mesin yang siap melayani banyak pengguna. Disaat *Load Balancer* menerima permintaan layanan dari user, maka permintaan tersebut akan diteruskan ke server utama. *Load Balancer* dengan pintar dapat menentukan server mana yang memiliki *Load* yang lebih rendah dan respons yang lebih cepat. Bahkan bias menghentikan akses ke server yang sedang mengalami masalah dan hanya meneruskannya ke server yang dapat memberikan layanan. Hal ini salah satu kelebihan yang umumnya dimiliki

*load balancer* , Sehingga layanan seolah olah tidak ada gangguan di mata pengguna.

### **2.6.1 Cara Kerja Load Balancing**

Pada dasarnya disaat *load balancer* menerima permintaan layanan dari pengguna, maka permintaan tersebut akan diteruskan ke *server* utama. Cara kerja pada *apache* sebagai *load balancing* sendiri adalah pada saat pengguna meminta halaman dari web dari server, *apache load balancing* akan menerima permintaan tersebut dan akan meneruskan permintaan dari pengguna menuju server-server yang menjadi anggota dari *apache load balancer*. (Supramana & Prisma: 2016)

### **2.6.2 Algoritma Load Balancing**

#### 1. Round Robin

Algoritma ini membagi beban secara berurutan dan merata dari satu server ke server lain

#### 2. Ratio

Algoritma dengan parameter diberikan untuk masing-masing server yang akan dimasukkan kedalam sistem *load balancing*. Server dengan ratio terbesar akan diberi beban besar sedangkan server dengan ratio kecil akan diberikan beban kecil.

#### 3. Fastest

Algoritma dengan mengutamakan server-server yang memiliki repon yang paling cepat pada saat permintaan masuk.

#### 4. Least connection

Algoritma membagi beban berdasarkan banyaknya koneksi yang sedang dilayani oleh server. Server yang memiliki koneksi paling sedikit akan melayani permintaan yang masuk. (Supramana & Prisma: 2016)

### 2.6.3 Tipe Load Balancing

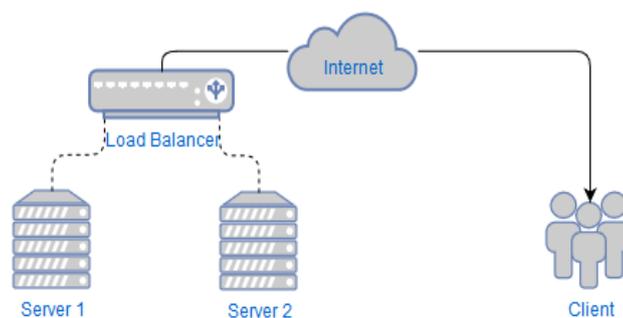
Dalam perancangan *load balancing* terdapat dua pilihan tipe *load balancer* yaitu sebagai berikut :

#### 1. Software Load Balancing

Load balancer yang sering digunakan berbasis perangkat lunak dimana *load balancing* berjalan disebuah *server* dan aplikasi *load balancing* di install dan perlu di konfigurasi sebelum dapat berfungsi. Proses *load balancing* dipengaruhi oleh perangkat yang digunakan seperti kartu jaringan (*Network Interface Card*), besarnya RAM (*Random Access Memory*) dan juga media penyimpanan.

#### 2. Hardware Load Balancing

Load balancer yang berjalan disebuah alat yang siap digunakan. Tipe *load balancing* ini banyak digunakan karena kemudahannya. (Supramana & Prisma: 2016)



**Gambar 2.4** Arsitektur Load Balancing

## 2.7 Pengertian Jaringan Komputer

Jaringan komputer adalah sekelompok komputer yang saling terhubung satu dengan yang lain melalui media kabel ataupun gelombang radio dengan tujuan untuk saling bertukar data. Pada dasarnya jaringan komputer dibangun oleh teknologi komputer dan teknologi komunikasi yang memiliki skala yang berbeda-

beda mulai dari jaringan LAN hingga jaringan internet. Tanpa kita sadari teknologi ini berkembang sangat pesat dan sangat membantu penggunanya, mulai dari hanya sekedar bertukar data hingga melakukan pembayaran secara online. Jaringan komputer juga dapat membantu berbagi resource yang dapat digunakan bersama-sama, sehingga dapat menekan biaya pengadaan alat.

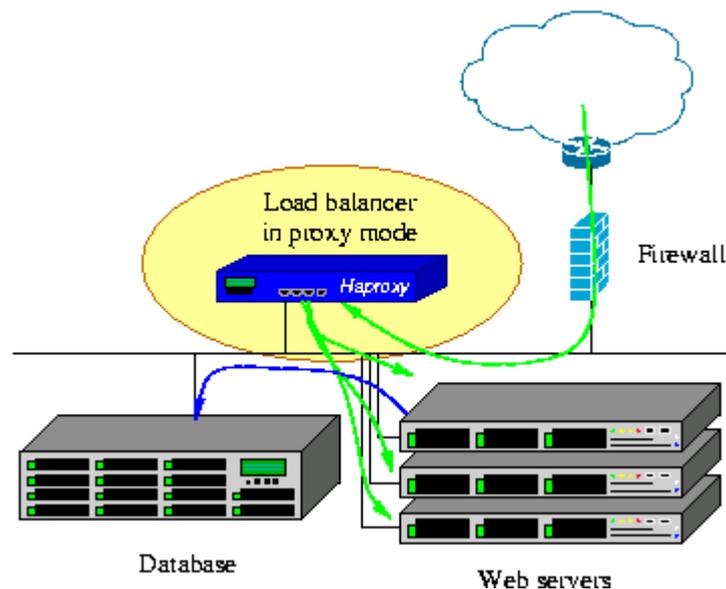
Berdasarkan kegunaannya klasifikasi jaringan dapat dibagi menjadi 2 macam yaitu jaringan client server dan jaringan peer to peer. Jaringan client server adalah jaringan yang salah satunya menjadi pusat dan pembantu komputer lain. Dalam prosesnya jaringan client server memiliki ciri khas yang mencolok yaitu setiap client melakukan request keperluan kepada server kemudian server akan menjawab sesuai request dan resource yang server miliki. Jaringan client server ini dalam skala besar dapat menghubungkan client dengan client yang berada di belahan bumi lain, contohnya adalah web streaming video atau media social. Jaringan server client ini memiliki beberapa keunggulan seperti kontrol yang terpusat sehingga server memiliki wewenang untuk mengontrol akses setiap client nya. Dibandingkan dengan jaringan peer to peer, jaringan client server memiliki kelemahan jika terjadi kegagalan pada pusat kontrol maka seluruh client tidak bias dilayani, terlalu banyak pengguna menjadikan jaringan client server lambat.

Jika jaringan client server menjadikan salah satu komputer menjadi pembantu berbeda dengan jaringan berbeda dengan jaringan peer to peer, pada jaringan peer to peer semua komputer yang saling terhubung bekerja sama dan saling berbagi resource dan service untuk menyelesaikan sebuah tugas. Setiap komputer dapat menggunakan aplikasi yang terdapat di komputer lain, komputer B kemudian dicetak langsung dari komputer A (Heronimus : 2019)

## **2.8 HAProxy**

HAProxy adalah produk *open source* yang mendukung keperluan penyeimbang beban dari failover web server, banyak digunakan untuk keperluan reverse proxy di site-site yang trafik hariannya tinggi. Pada HAProxy memiliki beberapa parameter didalamnya yaitu sebagai berikut.

1. Global parameter : berisi manajemen proses dan keamanan tuning kinerja, debugging dan userlist
2. Proxy
3. Konfigurasi server
4. Manipulasi HTTP.
5. Acceslist
6. Logging
7. Statistik dan monitoring.



Sumber : (<http://www.haproxy.org/>)

**Gambar 2.5** HAProxy

HAProxy adalah penyeimbang beban yang dapat memuat dan menyeimbangkan layanan TCP. Khususnya cocok untuk load balancing HTTP HAProxy sama seperti dengan Heartbeat hanya saja pada heartbeat tidak ada fasilitas berupa fitur statistic menggunakan webbased html.

Pada HAProxy ini juga sudah tersedia fitur algoritma Round Robin jadi untuk menggunakannya seseorang teknisi tinggal men-setting pada file konfigurasinya saja (Adnan : 2017)

## 2.9 Reverse Proxy

Terdapat beberapa jenis sistem penyeimbang beban (Load balancing) dan salah satu jenisnya merupakan proxy. Proxy adalah sebuah sistem komputer atau program aplikasi yang melayani permintaan dari klien dengan meminta layanan ke server lain.

Proxy server memiliki 3 fungsi utama yaitu :

1. *Connection sharing* : perantara client dan server.
2. *Filtering* : bekerja pada layer aplikasi yang dapat memblokir paket-paket tertentu.
3. *Caching* : mampu menyimpan informasi yang pernah diakses dari server-server

Proxy di bagi menjadi 2 yaitu *forward proxy* dan *reverse proxy*. *Forward proxy* adalah *proxy* yang meneruskan data ke host tujuan. *Reverse proxy* adalah sebuah *proxy* yang berada di depan web server, digunakan sebagai cache atau bias juga sebagai load balancer. *Reverse proxy* menjadi perantara user-user di internet terhadap akses ke web-server yang berada pada *local area network*, sehingga langsung web server yang dimaksud padahal sesungguhnya user di internet mengakses web-server yang terdapat di *local area network* melalui *reverse proxy* tersebut. (Supramana & Prisma: 2016)

## 2.10 Virtualisasi Server

Virtualisasi Server merupakan konsep baru pada perkembangan teknologi. Hal ini dikarenakan, virtualisasi server memungkinkan penggunaan satu perangkat keras untuk menjalankan beberapa sistem operasi secara independent dengan layanan yang berbeda pada waktu bersamaan. Virtualisasi server sebagai paradigma baru memberikan efisiensi yang lebih dibanding dengan sistem server yang ada dengan meminimalkan jumlah perangkat fisik, tempat, biaya pemeliharaan sistem. (Rosalia, Rendi & Ratna : 2016)