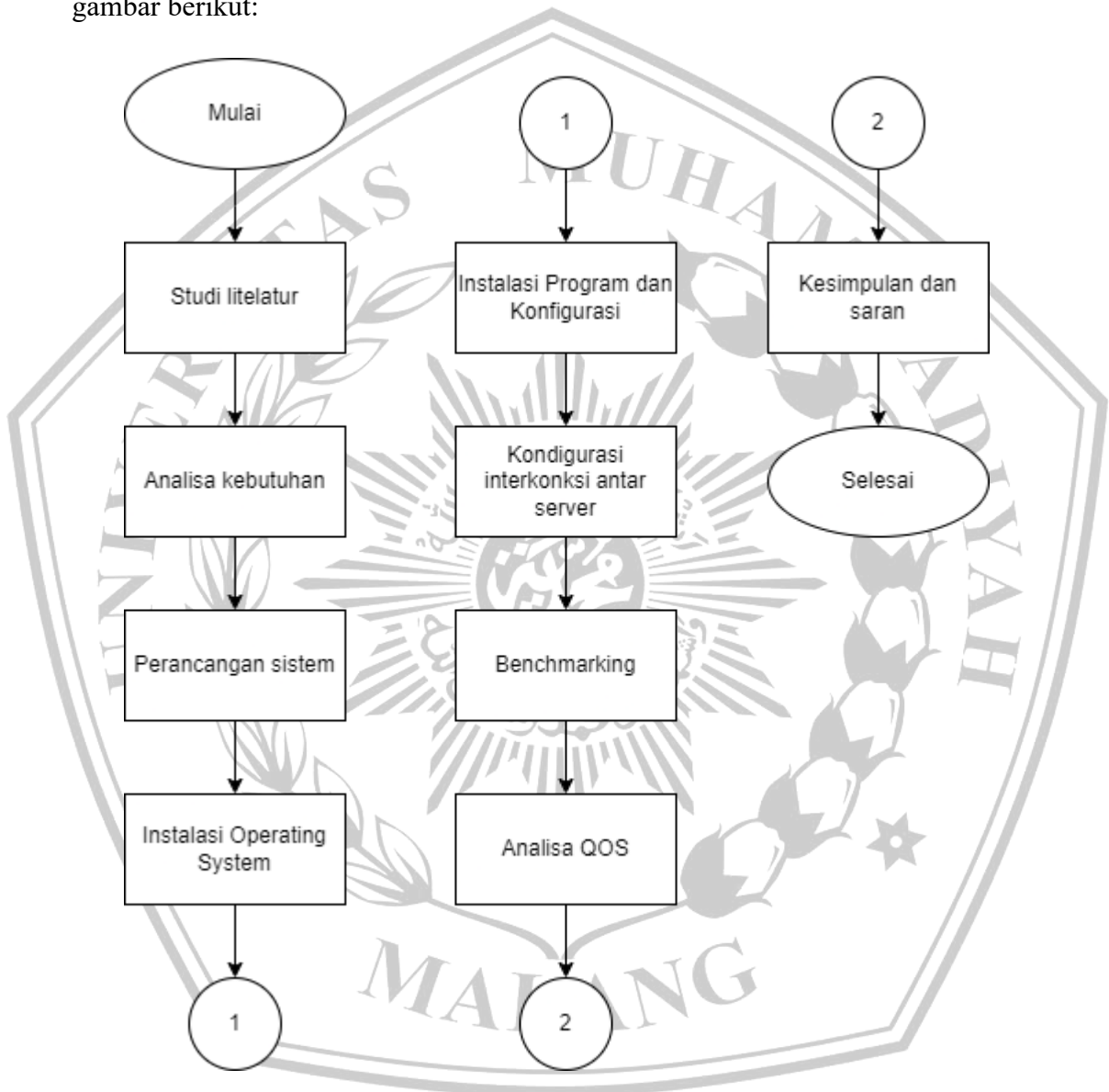


BAB III

METODOLOGI PENELITIAN

3.1 Metode

Dalam penyusunan skripsi ini, terdapat metodologi yang ditunjukkan pada gambar berikut:



Gambar 3.1 Flowchart perencanaan penelitian

3.2 Studi Litelatur

Untuk mencapai tujuan penelitian dan menyelesaikan masalah yang ada, peneliti melakukan studi literatur. Studi ini bertujuan untuk mempelajari teori-teori yang relevan, seperti load balancing dan algoritma yang dipakai. Peneliti menggunakan berbagai sumber referensi, seperti jurnal ilmiah dan e-book untuk mendukung penelitian ini.

3.3 Analisa Kebutuhan

Analisa untuk menentukan komponen yang diperlukan untk dapat menjalankan program Load Balancing secara keseluruhan melalui virtualisasi. Menentukan system operasi atau host yang digunakan baik dari host windows dan juga server virtualisasinya.

1. Perangkat Keras

Pada penelitian ini menggunakan komponen-komponen perangkat keras pada satu komputer yang memiliki spesifikasi sebagai berikut:

Tabel 3.1 Spesifikasi Komputer

Motherboard	Asrock B450 Pro4
CPU	Ryzen 5 5600 6 core 12 thread
GPU	MSI Ventus GTX 1660 OC 6GB
RAM	TeamGroup DDR4 16 GB
Storage	Adata XPG SX8200 PRO 1TB
PSU	FSP Hydro K 550

2. Perangkat Lunak

Pada penelitian ini menggunakan beberapa perangkat lunak seperti virtual machine berupa VMware dan juga system operasi yang digunakan mulai dari host untuk virtual machine hingga Operating System pada virtual server itu sendiri terdiri dari:

a. Di luar server

1. Windows 11 Ver. 23H2 (OS Build 22631.3737)
2. Debian Server 12.5.0

3. VMware® Workstation 17 Pro Ver. 17.5.1 build-23298084
4. Putty Ver. Release 0.78
5. Java JRE Ver. 8 Update 361
6. Apache JMeter Ver. 5.5

b. Di dalam server

1. Nginx
2. Apache2
3. Cisco Paket Tracer 8.2.2

3.4 Perancangan Sistem

Software utama yang digunakan pada penelitian ini adalah menggunakan VMware workstation 17 dengan versi 17.5.1. Dengan menggunakan VMware workstation, program dapat dibentuk sedemikian rupa untuk mereplika penggunaan di lapangan atau untuk dapat membuat server yang lebih fleksibel karena tiap virtual OS dapat diinstall dan dijalankan dibawah Host system windows.

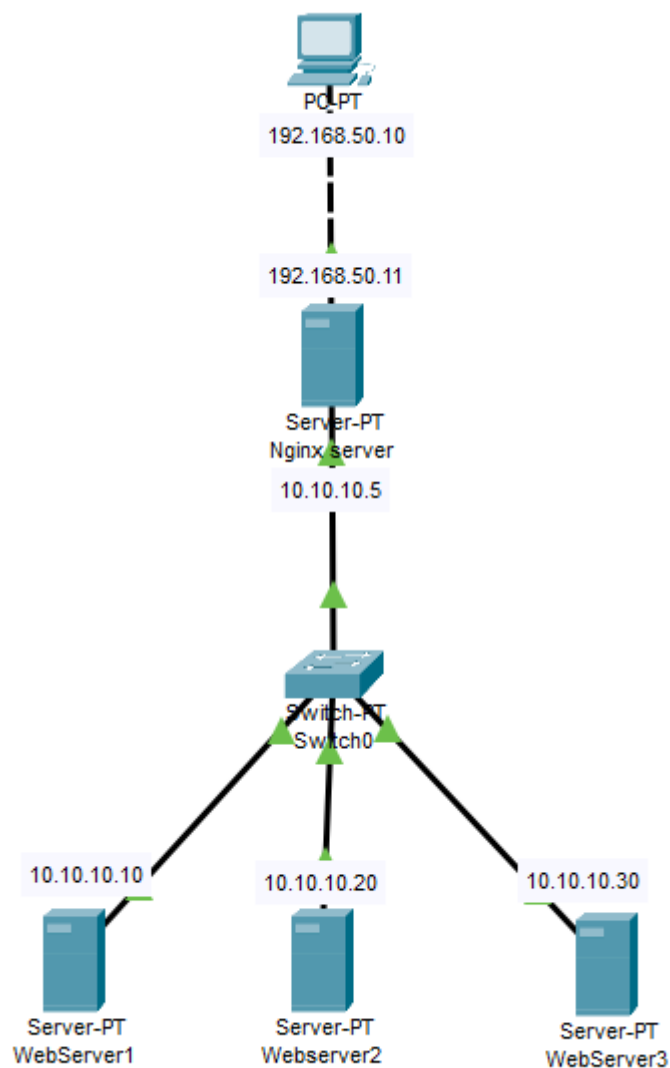
Operatis system windows yang digunakan adalah Windows 11 yang nantinya digunakan sebagai Host operating system. Spesifikasi dari Host Operating system itu sendiri dapat dilihat pada tabel. Dan versi dari windows yang digunakan pada host operating system adalah 23H2 dan dapat dilihat pada gambar sebagai berikut:



Gambar 3.2 Versi windows OS

Pembuatan topologi untuk menghubungkan antara Load Balance Server ke beberapa backend server. Alokasi ram per web server dan alokasi core atau thread per web server. Pembuatan interface untuk masing-masing web server dan juga Load balance server sesuai kebutuhan.

Dalam percobaan ini, digunakan rancangan jaringan lokal sederhana seperti yang ditunjukkan pada gambar di bawah ini:



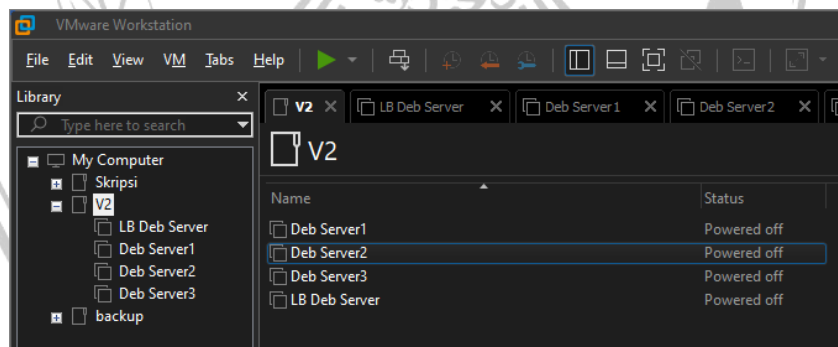
Gambar 3.3 Topologi jaringan

Berikut merupakan detail dari konfigurasi IP di masing-masing server dan komputer:

Tabel 3.2 Tabel alamat IP

Nama	IP Address	Keterangan
PC Tester	Eth0: 192.168.50.10	PC Host berupa windows
Nginx Load Balance Server	Eth0: 192.168.50.11 Eth1: 10.10.10.5	Server load balance, Eth0 digunakan untuk dapat diakses oleh PC tester, Eth1 untuk dapat berkomunikasi dengan backend server
Server1	Eth0: 10.10.10.10	Web Server 1
Server2	Eth0: 10.10.10.20	Web Server 2
Server3	Eth0: 10.10.10.30	Web Server 3

Kemudian instalasi virtual OS untuk Server Load Balance dan juga Web Server. Berikut 3 Virtual Server yang digunakan pada penelitian ini:

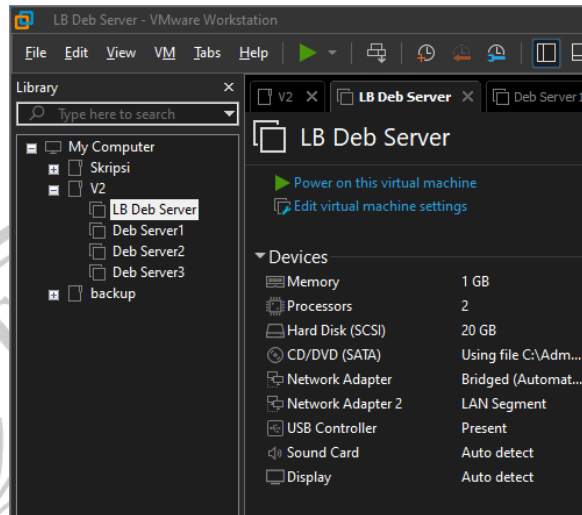


Gambar 3.4 Daftar virtual OS

1. Load Balance server

Server ini digunakan untuk membagi beban yang diberikan untuk web server yang ada di belakangnya. Kemudian untuk spesifikasi yang diberikan untuk Server Load balance adalah 2 inti prosesor, 1 GB RAM, 20 GB

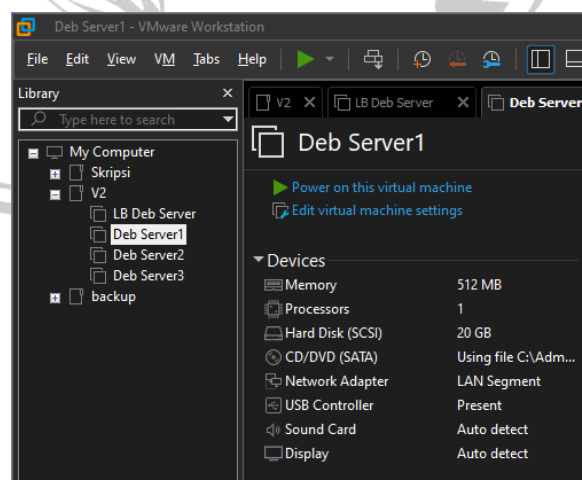
penyimpanan, dan 2 lan interface. Lan 1 berupa bridge network interface, Lan 2 berupa segment lan untuk keperluan komunikasi antara web server. Secara keseluruhan spesifikasi dari Load balance server dapat dilihat di bawah ini:



Gambar 3.5 Spesifikasi Load Balance server

2. Web Server 1

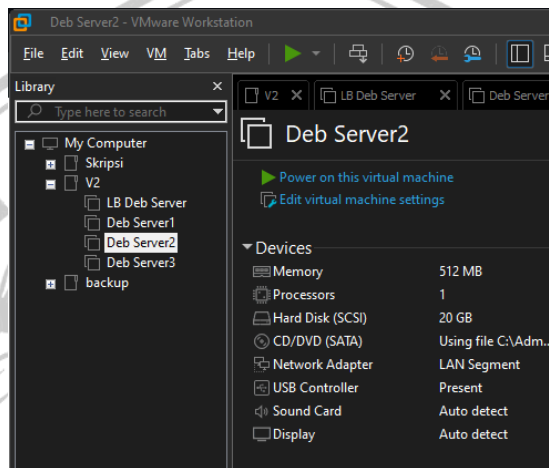
Server yang digunakan sebagai web server yang nantinya akan menjadi backend dari Load balance server memiliki spesifikasi yang lebih rendah dari Load balance Server. Yaitu 1 inti core, 512 MB RAM, 20 GB penyimpanan, dan 1 interface LAN yang digunakan untuk berkomunikasi antara server yang lain. Berikut adalah gambar dari spesifikasi dari web server 1:



Gambar 3.6 Spesifikasi Server 1

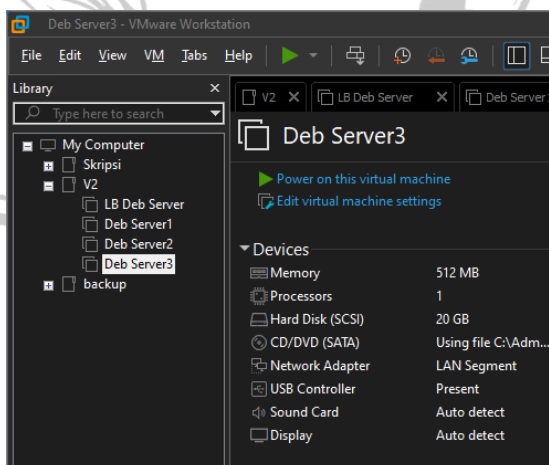
3. Web Server 2

Sama dengan Web server 1, server ini digunakan sebagai server backend yang berkerja beriringan dengan web server yang lain. Sama seperti spesifikasi dari web server 1, server ini memiliki spesifikasi sebagai berikut; 1 inti core, 512 MB RAM, 20 GB penyimpanan dan 1 interface LAN yang digunakan untuk berkomunikasi antara server yang lain. Berikut adalah gambar dari spesifikasi web server 2:



Gambar 3.7 Spesifikasi Server 2

4. Dan yang terakhir adalah web server 3, server backend yang memiliki spesifikasi yang sama dengan web server yang lainnya. Spesifikasi dari web server 3 yaitu; 1 inti prosesor, 512 MB RAM, 20 GB penyimpanan, dan 1 buah LAN interface untuk dapat berkomunikasi dengan server yang lain. Berikut gambar spesifikasi web server 3:



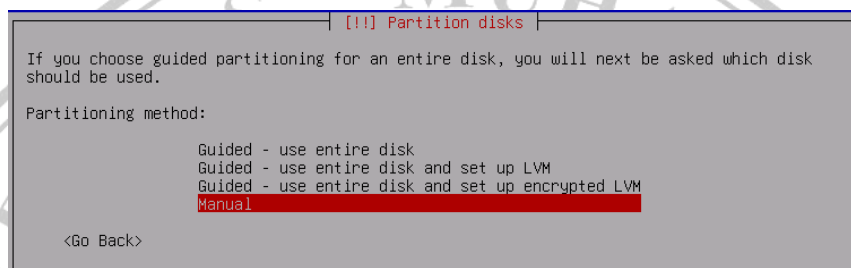
Gambar 3.8 Spesifikasi Server 3

3.5 Instalasi Sistem Operasi

Dalam pembuatan server diperlukan sebuah system operasi yang dapat berjalan dalam virtual machine yang nantinya akan diinstall sebuah program yang berguna untuk menjalankan fungsi dari server itu sendiri. Pada penelitian ini system operasi yang digunakan adalah Debian server untuk keseluruhan server yang digunakan. Berikut langkah – langkah dalam proses instalasi Debian server:

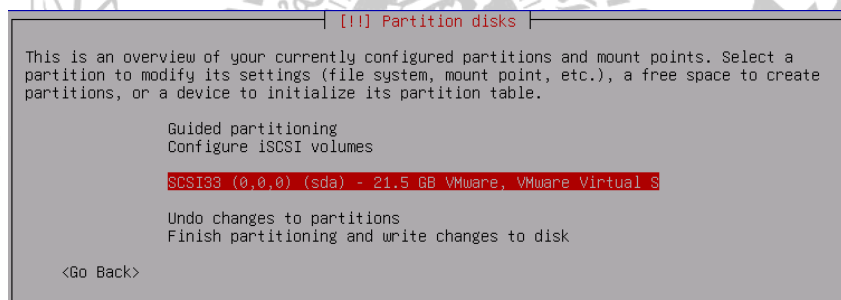
1. Membuat virtual machine baru yang diakses pada tab file, lalu klik New Virtual Machine
2. Kemudian pilih opsi Custom untuk mendapatkan pilihan pengaturan yang lebih lengkap
3. Pada Hardware Compability pilih pada pilihan yang terbaru
4. Kemudian pilih file iso dari Debian yang akan diinstall pada pilihan nomor urut dua
5. Kemudian beri nama pada virtual machine yang akan dibuat, sebagai contoh “Debian Base”. Kemudian tentukan di mana file Debian akan diinstall atau biarkan pada lokasi default.
6. Kemudian akan diminta berapa jumlah core yang dialokasikan untuk virtual machine yang akan dibuat. Dua core untuk server nginx load balance, dan satu core untuk web server
7. Selanjutnya ubah jumlah alokasi ram yang digunakan. 1 GB untuk nginx load balance server, dan 512 MB untuk web server
8. Selanjutnya mode yang digunakan pada interface dapat berupa NAT untuk terlebih dahulu. Terdapat dua interface pada nginx load balance, sedangkan untuk web server hanya perlu menggunakan 1 interface saja
9. Kemudian create new virtual disk
10. Kemudian pilih opsi “store a virtual disk as a single disk” dan gunakan size default dari berapa storage yang bisa digunakan
11. Kemudian next sampai setup selesai. Kemudian jalankan virtual machine yang telah dibuat
12. Setelah klik install, lalu pilih country region yang akan digunakan
13. Kemudian lanjut sampai Key map to use, gunakan default yaitu; American English

14. Selanjutnya untuk nama host yang digunakan dapat menggunakan defaultnya terlebih dahulu, yaitu; Debian. Karena nantinya akan diganti sesuai dengan peruntukan servernya
15. Domain bisa dikosongkan
16. Kemudian membuat root password, setelah mengisikan root password dua kali. Buat user baru dan juga passwordnya
17. Selanjutnya untuk timezone pilih opsi yang sesuai dari zona waktu di mana tempat berada
18. Selanjutnya untuk partisi yang akan digunakan, pilih opsi manual



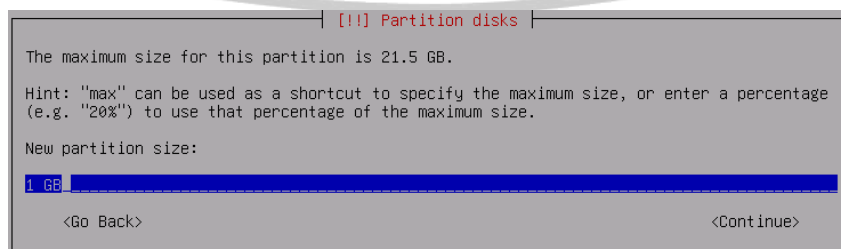
Gambar 3.9 Partisi manual

19. Kemudian pilih disk yang akan dipartisi



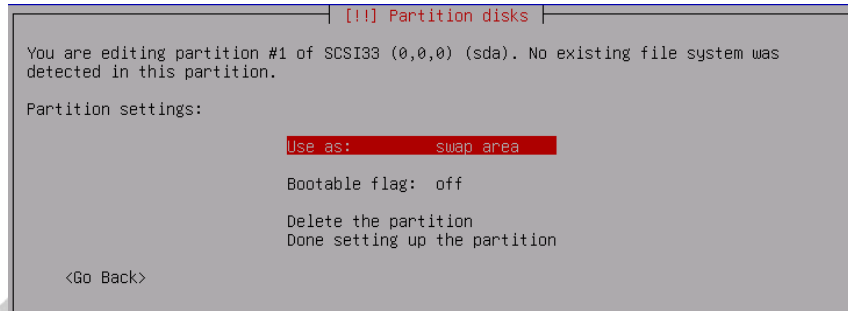
Gambar 3.10 Daftar virtual disk

20. Kemudian klik free space yang ada, kemudian klik create new partition
21. Partisi swap diperlukan agar server dapat berjalan dengan baik. Pada umumnya ukuran swap yang digunakan pada linux adalah menggunakan perhitungan dua kali kapasitas RAM yang ada.



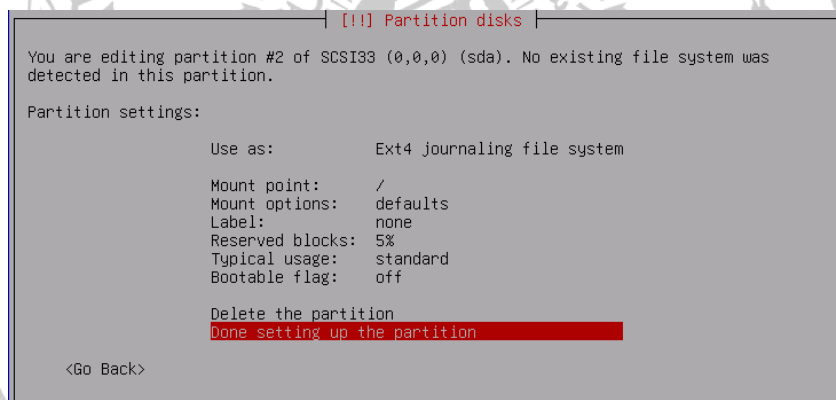
Gambar 3.11 Besar partisi swap

22. Kemudian pilih primary, kemudian location for the new partition kita pilih pada opsi; end. Karena agar nantinya partisi swap akan berada paling akhir di urutan partisi
23. Pada use as, pilih opsi swap area agar partisi swap dapat digunakan, kemudian klik done setting



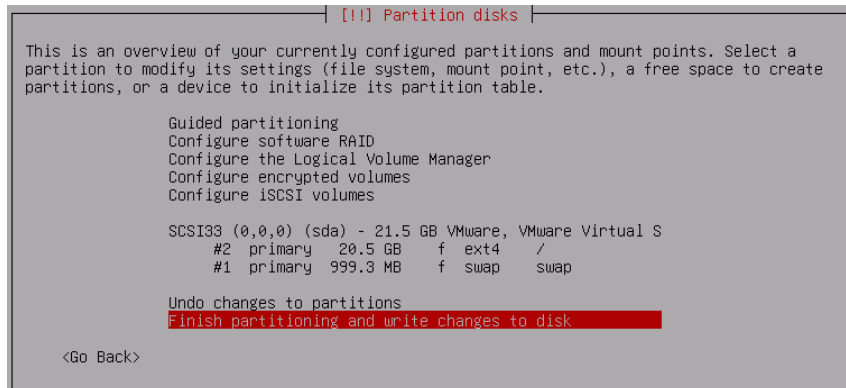
Gambar 3.12 Pengaturan partisi swap

24. Kemudian klik lagi pada free space yang ada, create a new partition. Gunakan semua sisa dari partisi yang belum terpakai. Pilih opsi primary.
25. Kemudian pada partition setting, set use as pada Ext4 journaling file system dan mount point pada: / (root)



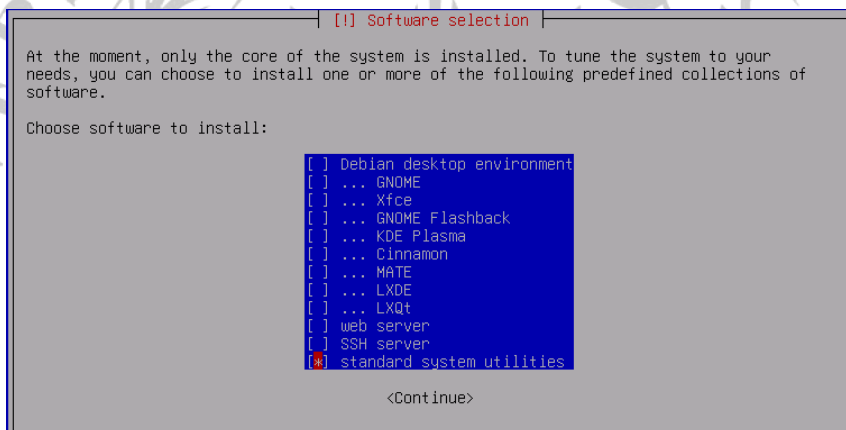
Gambar 3.13 Pengaturan partisi root

26. Setelah semua partisi selesai dibuat, berikut adalah tampilan dari semua partisi yang akan digunakan beserta atributnya



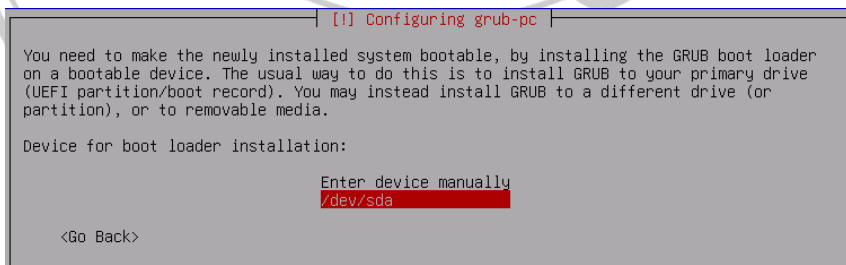
Gambar 3.14 Hasil akhir partisi

27. Setelah klik finish, lakukan simpan pengaturan dari partisi yang dibuat tadi dengan memilih opsi yes pada Write the change to disk
28. Kemudian untuk kebutuhan server, hilangkan centang pada Debian desktop environment dan gnome.



Gambar 3.15 Seleksi perangkat lunak

29. Proses instalasi berjalan dan pada bagian terakhir yaitu menginstall GRUB boot loader ke disk, pilih yes kemudian pilih disk yang ada.

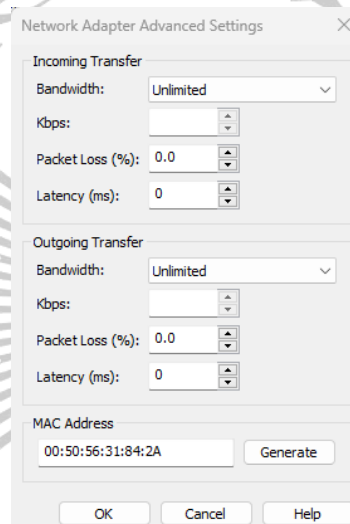


Gambar 3.16 Lokasi install master boot record

30. Pada tahap ini proses instalasi telah selesai kemudian system akan reboot.

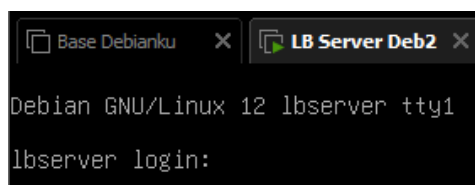
Selanjutnya adalah proses duplikasi virtual machine yang telah dibuat, hal ini dilakukan untuk menghemat waktu dikarenakan proses instalasi yang melalui beberapa tahap dan juga konfigurasi yang ada di dalamnya. Yang harus diperhatikan adalah nama virtual machine baru yang telah diduplikasi dan juga hostnamanya harus diganti untuk memudahkan mengidentifikasi virtual machine. Berikut adalah langkah- langkah dalam melakukannya:

1. Pastikan Virtual machine dalam keadaan mati, kemudian lakukan full clone pada virtual machine tersebut dan beri nama virtual machine yang baru sesuai kebutuhan server
2. Setelah selesai cloning, hal berikutnya yang harus dilakukan adalah melakukan generate mac address yang digunakan pada LAN interface.



Gambar 3.17 Generate alamat MAC

3. Kemudian nama dari hostname masih sama dengan virtual machine sebelumnya. Makah hal selanjutnya yang harus dilakukan adalah mengubah nama dari hostname virtual machine tersebut
4. Login, kemudian edit nama hostname dengan `#nano /etc/hostname`
5. Selanjutnya edit `#nano /etc/hosts`
6. Kemudian reboot, setelah itu bisa dilihat nama hostname telah berubah



Gambar 3.18 Nama host

3.6 Instalasi Program dan Konfigurasi

Adapun program beserta konfigurasi yang harus dilakukan pada tiap server yang harus dilakukan karena tiap server memiliki fungsi dan kegunaan yang berbeda. Berikut adalah program dan konfigurasi yang digunakan dalam server:

a. Server Load Balance

Pada server load balance, program yang digunakan adalah NGINX, berikut adalah langkah proses instalasi program beserta konfigurasinya:

1. Update repository linux terlebih dahulu disaat linux baru selesai diinstall. Dapat dilakukan dengan #apt update

```
root@lserver:~# apt update
Get:1 http://security.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Hit:2 http://kartolo.sby.datautama.net.id/debian bookworm InRelease
Get:3 http://kartolo.sby.datautama.net.id/debian bookworm-updates InRelease [55.4 kB]
Fetched 103 kB in 0s (247 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@lserver:~#
```

Gambar 3.19 Update repository

2. Kemudian install program NGINX dengan perintah #apt install nginx
3. Kemudian buat konfigurasi untuk menjadikan nginx sebagai load balancer. Buat file ber-ekstensi .conf pada /etc/nginx/sites-available. Sebagai contoh #nano /etc/nginx/sites-available/mylb.conf. Lalu buat server backend berdasarkan ip web server yang telah ditentukan

```
LB Deb Server x
GNU nano 7.2 /etc/nginx/sites-available/mylb.conf *
upstream web-backend {
server 10.10.10.10:80;
server 10.10.10.20:80;
server 10.10.10.30:80;
}

server {
listen 80;
location / {
proxy_pass http://web-backend;
}
}
```

Gambar 3.20 Konfigurasi nginx load balance

4. Kemudian binding atau buat link antara file konfigurasi yang ada di sites-available dengan sites-enabled

```
root@lserver:~# ln -s /etc/nginx/sites-available/mylb.conf /etc/nginx/sites-enabled/
```

Gambar 3.21 Binding file konfigurasi

5. Kemudian lakukan test pada konfigurasi nginx yang telah dilakukan dengan `#nginx -t`. Jika berhasil maka akan muncul seperti gambar berikut

```
root@lvserver:~# nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
root@lvserver:~# _
```

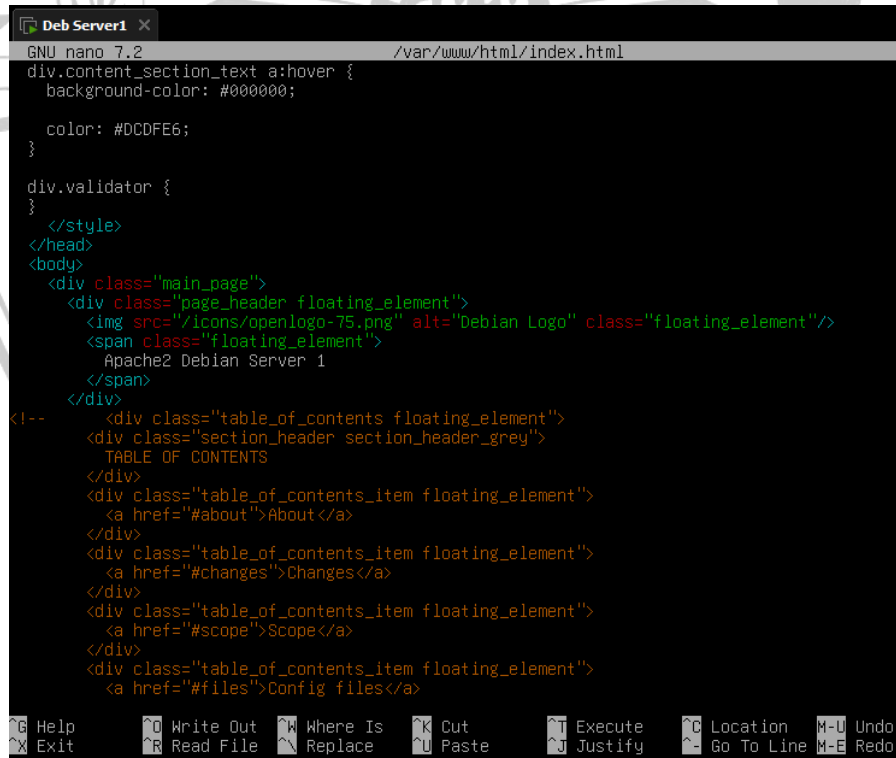
Gambar 3.22 Test konfigurasi nginx

6. Kemudian restart nginx `#systemctl restart nginx`

b. Web Server

Pada web server, program yang digunakan berupa apache web server. Untuk proses instalasi dan konfigurasi sebagai berikut

1. Update repository dengan `#apt update`
2. Kemudian install apache web server `#apt install apache2`
3. Lalu edit konten yang ada di `/var/www/html/index.html`. Ubah sehingga antara web server satu dengan lainnya tidak sama untuk memudahkan identifikasi web server nantinya



```
GNU nano 7.2 /var/www/html/index.html
div.content_section_text a:hover {
    background-color: #000000;

    color: #DCDFE6;
}

div.validator {
}
</style>
</head>
<body>
  <div class="main_page">
    <div class="page_header floating_element">
      
      <span class="floating_element">
        Apache2 Debian Server 1
      </span>
    </div>
  <!--
    <div class="table_of_contents floating_element">
      <div class="section_header section_header_grey">
        TABLE OF CONTENTS
      </div>
      <div class="table_of_contents_item floating_element">
        <a href="#about">About</a>
      </div>
      <div class="table_of_contents_item floating_element">
        <a href="#changes">Changes</a>
      </div>
      <div class="table_of_contents_item floating_element">
        <a href="#scope">Scope</a>
      </div>
      <div class="table_of_contents_item floating_element">
        <a href="#files">Config files</a>
      </div>
  </--
  </body>
</html>
```

Gambar 3.23 Isi index.html

c. ApacheJmeter

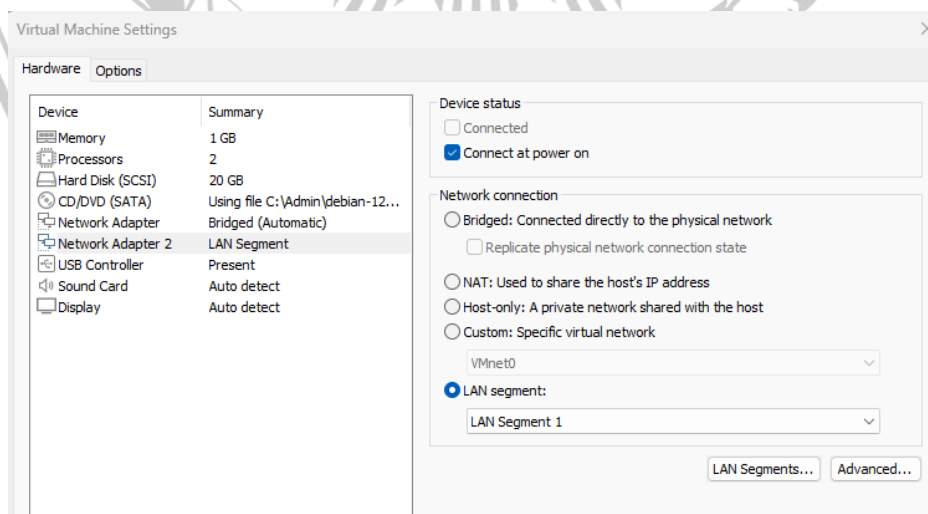
Software selanjutnya adalah ApacheJmeter. Software ini bergungsi untuk melakukan benchmark pada peneliian ini. Sebelum dapat digunakan, apache jmeter membutuhkan java untuk dapat berjalan. Setelah java terinstall, apachejmeter dapat dijalankan dengan mengeksekusi jmeter.bat

3.7 Konfigurasi interkoneksi antar server

Setelah instalasi system opearsi, program, dan konfigurasi selesai. Hal selanjutnya yang dilakukan adalah membuat koneksi atar server agar dapat saling berkomunikasi antara yang lainnya. Karena tiap server memiliki kegunaan yang berbeda, maka aka ada sedikit perbedaan dalam konfigurasinya. Untuk interkoneksi antar server, dibuatlah LAN Segment yang nantinya semua server akan memakai LAN segment yang sama.

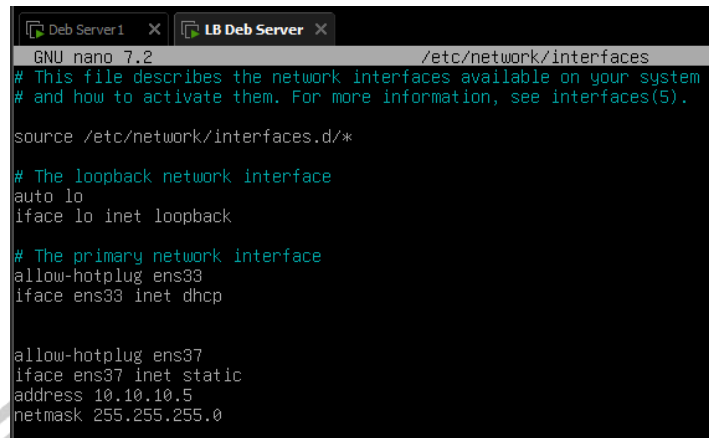
a. Konfigurasi Load Balance Server

Pada server Load Balance, dikarenakan server juga harus bisa berkomunikasi dengan host windows yang dimana ApacheJmeter dijalankan, maka diperlukan dua buah interface. Di sini interface yang digunakan berupa Bridge untuk interface satu, sedangkan interface dua berupa LAN Segment.



Gambar 3.24 Konfigurasi Lan Load balance server

Kemudian untuk konfigurasi IP bisa dilihat pada gambar di bawah:



```
GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

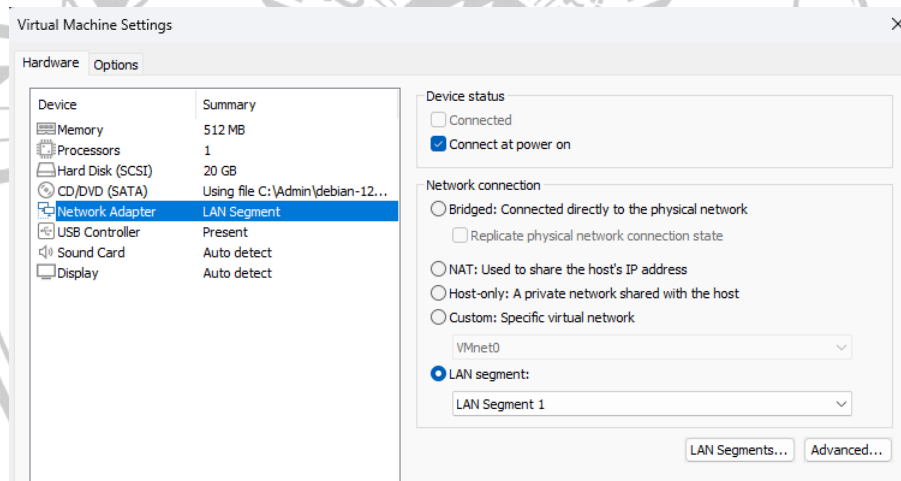
# The primary network interface
allow-hotplug ens33
iface ens33 inet dhcp

allow-hotplug ens37
iface ens37 inet static
address 10.10.10.5
netmask 255.255.255.0
```

Gambar 3.25 Konfigurasi alamat IP Load balance server

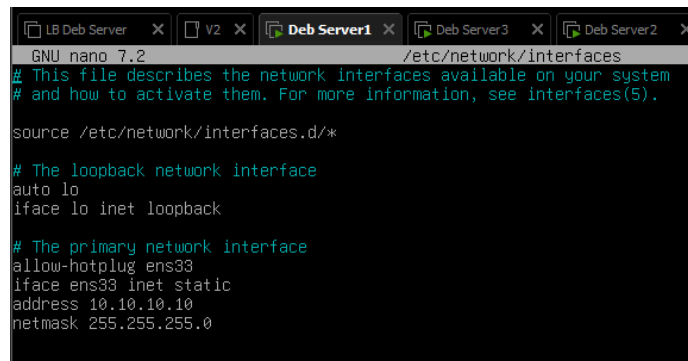
b. Konfigurasi Web Server

Karena web server hanya memiliki satu buah interface saja. Maka ketiga web server memiliki konfigurasi interface yang sama, dan dapat dilihat sebagai berikut:



Gambar 3.26 Konfigurasi Lan Web server

Dan konfigurasi IP dari tiap web server dapat dilihat sebagai berikut:



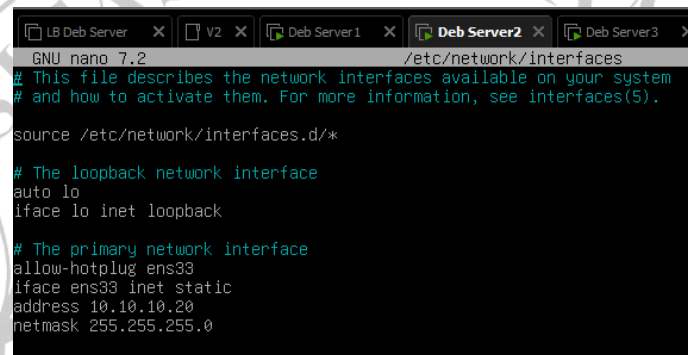
```
GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 10.10.10.10
netmask 255.255.255.0
```

Gambar 3.27 Konfigurasi alamat IP Web server 1



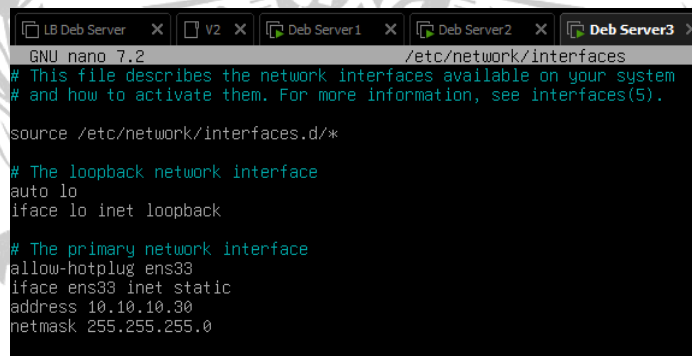
```
GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 10.10.10.20
netmask 255.255.255.0
```

Gambar 3.28 Konfigurasi alamat IP Web server 2



```
GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 10.10.10.30
netmask 255.255.255.0
```

Gambar 3.29 Konfigurasi alamat IP Web server 3

3.8 Benchmarking

Benchmarking menggunakan program yang bernama ApacheJmeter. Dengan menjalankan scenario pengujian dan mengatur parameter pengujian. Dan juga memonitoring metrik performa seperti throughput, respons time, dan error rate selama pengujian.

3.9 Analisa QOS

Data yang diperoleh dari proses benchmarking akan dianalisis secara menyeluruh untuk mendapatkan gambaran yang komprehensif tentang performa system. Analisis ini akan berfokus pada metrik – metrik yang diukur selama benchmarking, seperti Throughput Delay, Jitter dan Response Time.

3.10 Kesimpulan dan Saran

Setelah menyelesaikan seluruh tahapan penelitian, langkah terakhir yang harus dilakukan adalah menyusun kesimpulan. Kesimpulan ini harus memuat jawaban atas tujuan penelitian yang telah ditetapkan di awal tugas akhir.

