

BAB 3

METODE PENELITIAN

3.1 ALUR PENELITIAN

Pada metode penelitian membahas mengenai tahap-tahap atau proses penelitian yang dilakukan. Dimulai dari mencari dan membaca literatur referensi penelitian, proses perancangan *virtual machine server/client*, dan menentukan skenario pengujian. Kemudian pemasangan sistem operasi, *web server*, *website dummy* yang diuji, dan *benchmark tool* Siege di sisi *client*. Setelah itu dilakukan simulasi untuk memeriksa seluruh perangkat yang sudah dipasang bekerja dengan baik. Lalu proses pengujian dilakukan untuk mengumpulkan data yang akan dianalisis.

Untuk mempermudah pembaca memahami alur kerja dari penelitian ini, dapat dilihat pada Gambar 3.1 yang secara sederhana menunjukkan langkah-langkah atau alur kerja dari setiap tahap penelitian yang dilakukan. Berikut merupakan penjelasan dari setiap tahap alur penelitian pada Gambar 3.1.

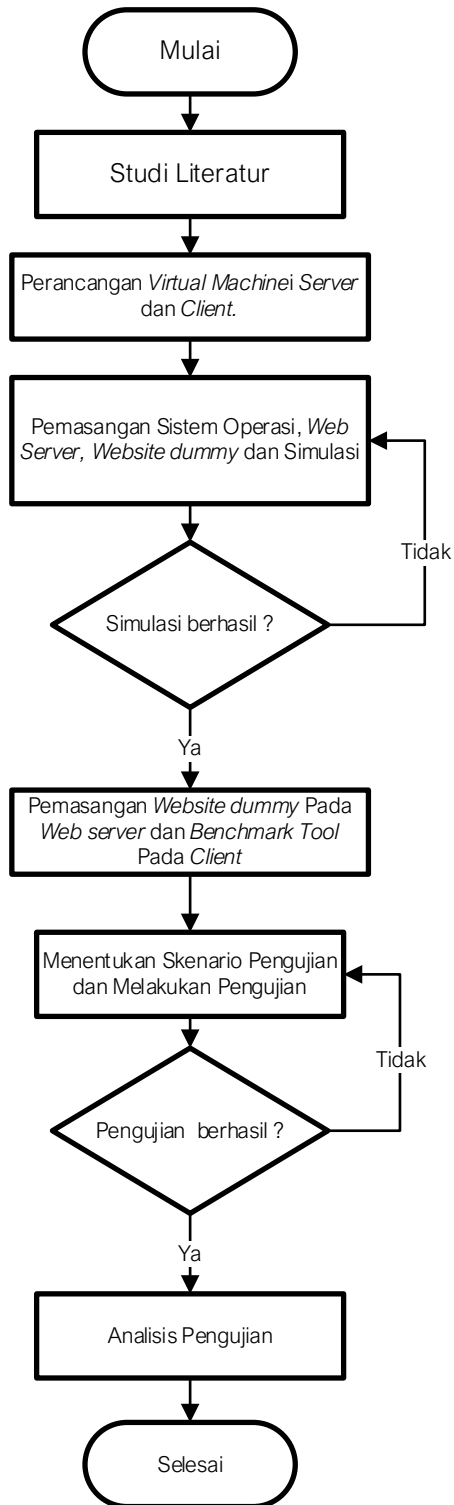
a. Studi Literatur

Tahap mencari dan mengumpulkan sumber materi yang dapat menjadi referensi penelitian dan memahami penelitian-penelitian sebelumnya yang berkaitan. Materi diperoleh dari berbagai sumber seperti jurnal, media, forum diskusi, dan buku.

b. Perancangan *Virtual machine Server Dan Client*

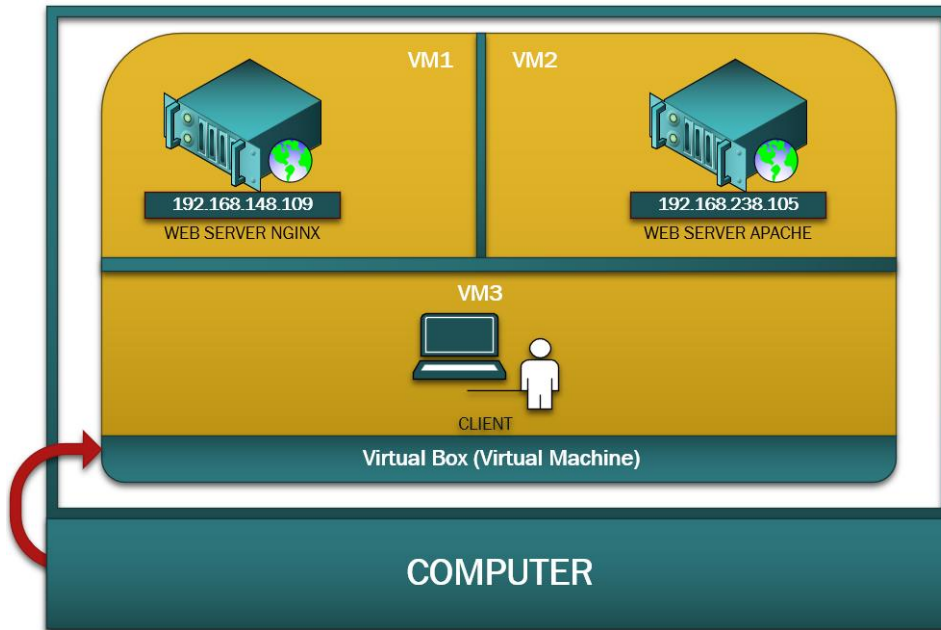
Tahapan ini membuat tiga *virtual machine* menggunakan perangkat lunak *Virtual Box* sebagai pengganti perangkat fisik asli yang berfungsi sebagai *server* dan *client*. Tiga *virtual machine* dibuat dengan pembagian dua *virtual machine* masing – masing sebagai *web server* Apache dan Nginx, kemudian satu *virtual machine* sebagai *client* yang berfungsi sebagai perangkat yang menguji kedua *web server*. Spesifikasi pada setiap *virtual machine* disamakan agar kemampuan setiap *virtual machine* pada masing – masing *web server* seimbang.

- c. Pemasangan Sistem Operasi, *Web server*, dan Simulasi.
Setelah *virtual machine* untuk *server* dan *client* telah dibuat, dilakukan instalasi sistem operasi linux Ubuntu *Server* pada *virtual machine* yang berfungsi sebagai *server* lalu diikuti dengan instalasi *web server* Apache dan Nginx. Di sisi *virtual machine* yang berfungsi sebagai *client*, dilakukan instalasi sistem operasi linux Ubuntu *Desktop*. Setelah instalasi yang sudah dilakukan berhasil, maka simulasi dilakukan untuk memeriksa fungsi setiap perangkat bekerja dengan baik. Simulasi yang dilakukan seperti menggunakan perangkat *client* untuk mengakses halaman *default* dari *web server* Apache dan Nginx.
- d. Pemasangan *website dummy* pada *web server* dan *Benchmark Tool* pada *Client*
Website dummy ditambahkan pada kedua *web server* sebagai bahan pengujian yang lebih bervariasi daripada menguji halaman *default* saja. *Website* ini terdiri atas beberapa halaman *web*. Halaman utama *website* ini akan menampilkan halaman *index*. Pada halaman *index* tersebut, terdapat beberapa menu yang mengarah ke halaman *web* yang lain. Setiap halaman *web* memiliki URL masing-masing. *Benchmark tool* *Siege* digunakan pada sisi *client* sebagai alat untuk menguji performa kedua *web server*.
- e. Menentukan Skenario Pengujian Dan Melakukan Pengujian
Skenario pengujian ditentukan untuk mendapatkan nilai parameter yang diinginkan. Penentuan skenario pengujian berdasarkan studi literatur dan referensi jurnal. Kemudian dilakukan pengujian berdasarkan skenario yang telah dibuat.
- f. Analisis Pengujian
Setelah pengujian dilakukan berdasarkan skenario yang telah ditentukan, maka akan didapatkan hasil data yang dibutuhkan berupa nilai parameter. Nilai parameter tersebut akan diubah kedalam bentuk tabel ataupun grafik untuk memudahkan analisis dari performa kedua *web server* yaitu Apache dan Nginx.



Gambar 3.1 Diagram Alir Penelitian

Pada penelitian ini, komunikasi antara *client* dengan *web server* Apache atau Nginx terhubung melalui internet seperti yang ditunjukkan pada Gambar 3.2.



Gambar 3.2 Topologi *Client-Web Server*

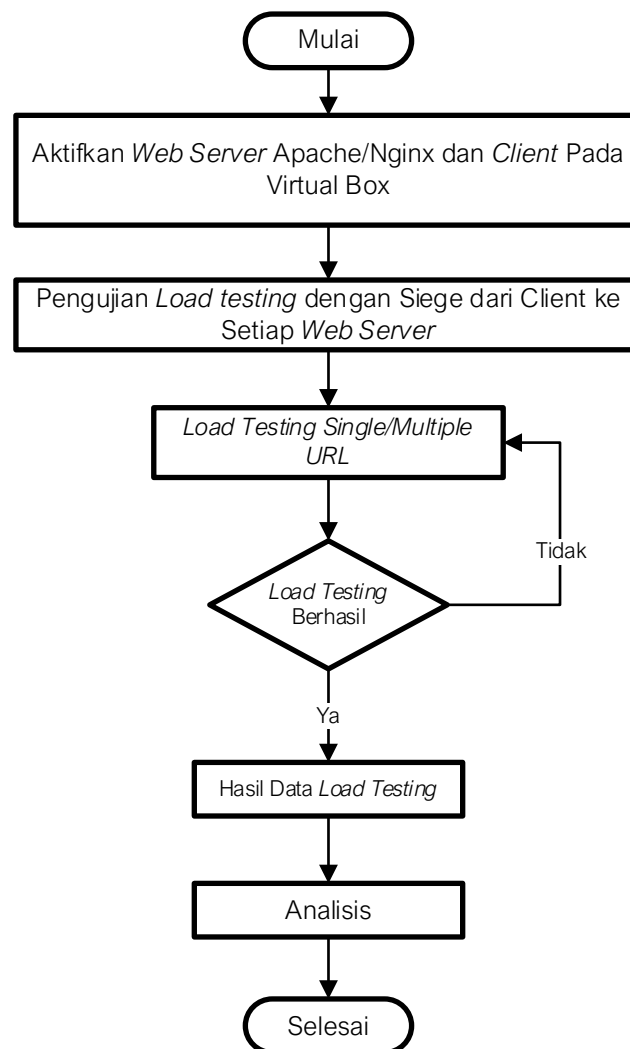
Pada Gambar topologi di atas menunjukkan 3 *virtual machine* yaitu pada satu komputer. *Web server* Apache atau Nginx memiliki IP tertentu yang dapat diakses oleh *client*.

3.2 ALUR SIMULASI/SKENARIO PENGUJIAN

Penelitian ini melakukan simulasi jaringan antara *client* dengan *web server* dengan memanfaatkan teknologi virtualisasi untuk dapat mengukur dan mengetahui performa dari *web server* berbasis Apache dan Nginx. Pengujian dilakukan dengan *benchmarking tool* Siege menggunakan metode *load testing* yang mensimulasikan sejumlah *virtual user* yang dapat dikonfigurasi untuk mengakses (*hit*) satu atau lebih halaman *web* yang diinginkan secara bersamaan. Performa *web server* dalam menangani berbagai *load condition* akan direkam dalam bentuk nilai parameter. Diagram alir simulasi dan pengujian pada penelitian ini ditunjukkan pada Gambar 3.3.

Dapat dilihat pada Gambar 3.3, dimulai dengan menyalakan *web server* Apache atau Nginx dan *Client* pada Virtual Box. Aktifkan salah satu *web server* sesuai dengan *web server* mana yang ingin diuji terlebih dahulu. Setelah salah satu *web server* aktif dan berjalan dengan baik, lakukan pengujian *load testing* menggunakan Siege dari sisi *client*. *Load testing* dilakukan dengan dua cara yaitu

load testing single Uniform Resource Locator (URL) dan load testing multiple Uniform Resource Locator (URL). Load testing single URL merupakan pengujian load testing yang dilakukan dengan cara mensimulasikan virtual user untuk mengakses satu halaman web atau URL. Load testing multiple URL merupakan pengujian load testing yang dilakukan dengan cara mensimulasikan virtual user untuk mengakses beberapa halaman web atau URL. Setelah load testing berhasil, akan didapatkan hasil data berupa nilai parameter yang akan dianalisis untuk mengetahui kinerja kedua web server. Pada tabel 3.1 disajikan beberapa skenario pengujian yang dilakukan berdasarkan beberapa load condition.



Gambar 3.3 Diagram Alir Simulasi dan Pengujian

Pengujian *load testing* pada setiap kondisi beban dilakukan sebanyak 10 kali, kemudian diambil rata-rata nilai parameter kinerja yang dihasilkan dari 10 kali pengujian tersebut. Tabel 3.1 menunjukkan skenario pengujian berdasarkan variabel pengujian yang dilakukan pada *load testing single URL* dan pada *load testing multiple URL*.

Tabel 3.1 Skenario Pengujian

Variabel Pengujian	<i>Load testing Single URL</i>	<i>Load testing Multiple URL</i>
<i>Concurrency</i>	10, 50, 100, 200	10, 50, 100, 200
<i>Repetition</i>	10, 100	-
<i>Delay (second)</i>	1	1
<i>Internet (Feature)</i>	-	Ya
<i>Time (minute)</i>	-	1, 3, 5, 10
<i>File</i>	index.html, profil.html, kontak.html	urls.txt

Variabel pengujian pada *load testing single URL* menggunakan *concurrency*, *repetition*, *delay*, kemudian diakhiri dengan URL / *file* yang ingin diuji. Variabel *concurrency*, *repetition* dan *delay* dikombinasikan untuk mendapatkan variasi kondisi beban. Variabel pengujian *internet* tidak digunakan pada *load testing single URL* karena variabel ini digunakan agar *virtual user* mengakses salah satu dari beberapa halaman *web* yang akan diuji dari *file* yang berisi kumpulan halaman *web* seperti yang digunakan pada *load testing multiple URL*. Pengujian *load testing single URL* ditujukan untuk mengukur seberapa cepat dan handal suatu *web server* dalam melayani sejumlah permintaan yang sama sehingga variabel *time* yang fungsinya untuk mengatur durasi pengujian tidak diperlukan.

Variabel pengujian pada *load testing multiple URL* menggunakan *concurrency*, *delay*, *time*, *internet* dan *file*. Variabel *concurrency*, *delay* dan *time* dikombinasikan untuk mendapatkan variasi kondisi beban. Pengujian *load testing multiple URL* ditujukan untuk mengukur kinerja *web server* dalam melayani

sejumlah permintaan yang berbeda-beda dalam durasi waktu tertentu sehingga variabel *repetition* yang berfungsi untuk mengulang permintaan tidak diperlukan.

Berdasarkan Tabel 3.1, berikut deskripsi skenario pengujian yang dilakukan.

- a) *Load condition* dan perintah pada saat melakukan *load testing single URL*
- 1) Siege -c10 -r10 -d1 --URL
 - 2) Siege -c10 -r100 -d1 --URL
 - 3) Siege -c50 -r10 -d1 --URL
 - 4) Siege -c50 -r100 -d1 --URL
 - 5) Siege -c100 -r10 -d1 --URL
 - 6) Siege -c100 -r100 -d1 --URL
 - 7) Siege -c200 -r10 -d1 --URL
 - 8) Siege -c200 -r100 -d1 --URL

Gambar 3.4 menunjukkan contoh perintah/*command* yang digunakan pada saat melakukan *load testing single URL*.

```
root@ubuntu-VirtualBox:~/siege# siege -c10 -r10 -d1 http://192.168.148.109/index.html
** SIEGE 4.0.4
** Preparing 10 concurrent users for battle.
The server is now under siege...
```

Gambar 3.4 Contoh Perintah *Load Testing Single URL*

- b) *Load condition* dan perintah pada saat melakukan *load testing multiple URL*
- 1) Siege -c10 -i -t1m -d1 --file urls.txt
 - 2) Siege -c50 -i -t3m -d1 --file urls.txt
 - 3) Siege -c100 -i -t5m -d1 --file urls.txt
 - 4) Siege -c200 -i -t10m -d1 --file urls.txt

Gambar 3.5 menunjukkan contoh perintah/*command* yang digunakan pada saat melakukan *load testing single URL*.

```
root@ubuntu-VirtualBox:~/siege# siege -c10 -i -t1m -d1 -l --file urls.txt
** SIEGE 4.0.4
** Preparing 10 concurrent users for battle.
```

Gambar 3.5 Contoh Perintah *Load Testing Multiple URL*

Berikut URL atau halaman *web* yang akan digunakan untuk melakukan pengujian *load testing single URL*.

- a) `http://192.168.238.152/index.html`
- b) `http://192.168.238.152/profil.html`
- c) `http://192.168.238.152/kontak.html`

Pada pengujian *load testing multiple URL*, *virtual user* akan mengakses beberapa URL atau halaman *web*. Oleh karena itu, beberapa URL tersebut dikumpulkan di dalam satu file bernama “urls.txt”. *Website dummy* yang digunakan pada pengujian ini memiliki 9 halaman *web* dengan halaman index sebagai halaman utama. Selain halaman utama, terdapat beberapa halaman *web* yang dapat dilihat pada lampiran seperti halaman profil, kontak, full-width, full-width2, full-width3, laporan, laporan2, dan laporan3. Semua halaman tersebut dikumpulkan di dalam file urls.txt seperti yang terlihat pada Gambar 3.6.

```
GNU nano 4.8                               urls.txt
http://192.168.148.109/index.html
http://192.168.148.109/profil.html
http://192.168.148.109/kontak.html
http://192.168.148.109/pages/full-width.html
http://192.168.148.109/pages/full-width2.html
http://192.168.148.109/pages/full-width3.html
http://192.168.148.109/pages/laporan.html
http://192.168.148.109/pages/laporan2.html
http://192.168.148.109/pages/laporan3.html
```

Gambar 3.6 File Konfigurasi *Multiple URL*

3.3 PERANGKAT YANG DIGUNAKAN

Penelitian ini dilakukan dalam bentuk simulasi untuk mendapatkan datanya. Simulasi tersebut membutuhkan berbagai alat penunjang agar dapat berjalan dengan baik. Alat penunjang yang digunakan berupa perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.3.1 Perangkat Keras (*Hardware*)

Perangkat keras merupakan perangkat fisik yang digunakan dalam penelitian. Dalam penelitian ini digunakan satu buah laptop.

1) Laptop

Penelitian ini menggunakan satu buah laptop yang menggunakan sistem operasi Windows 11 64-Bit. Laptop tersebut nantinya akan dipasang aplikasi *virtual machine*. Spesifikasi laptop yang digunakan dalam penelitian dapat dilihat pada tabel 3.1.

Tabel 3.2 Spesifikasi Laptop

Jenis Spesifikasi	Keterangan
Processor	AMD Ryzen 7 4800H 2.90 GHz
<i>Random Access Memory</i> (RAM)	16 GB DDR4 3200 MHz
<i>Storage</i>	SSD 512GB
<i>Network Interface Card</i> (NIC)	Intel Wi-Fi 6 AX200 160 MHz

3.3.2 Perangkat Lunak (*Software*)

Terdapat beberapa perangkat lunak yang digunakan dalam penelitian ini seperti Virtual Box, sistem operasi Ubuntu *Server*, sistem operasi Ubuntu *Dekstop*, dan *Siege*.

1) Virtual Box versi 6.1.30

Aplikasi Virtual Box digunakan untuk membuat *virtual machine* yang akan berfungsi sebagai *server* virtual menggantikan *server* fisik. Pada Virtual Box akan dibuat tiga *virtual machine* dengan spesifikasi yang tertera pada tabel 3.2. *Virtual machine 1* akan difungsikan sebagai *web server* berbasis Nginx dan *virtual machine 2* akan difungsikan sebagai *web server* berbasis

Apache. Selain itu, terdapat *virtual machine 3* yang berfungsi sebagai *client* yang akan melakukan pengujian menggunakan perangkat lunak Siege.

Tabel 3.3 Spesifikasi *Virtual Machine*

<i>Virtual machine</i>	Spesifikasi
<i>Virtual machine 1 &2</i>	<p>VM1 Name : Ubuntu Server Nginx Skripsi VM2 Name : Ubuntu Server Apache Skripsi Operating System : Ubuntu Server 21.10 (64 bit) Base Memory : 2 GB Processors : 2 Core Video Memory : 16 MB Storage : 30 GB Web server VM1 : Nginx Web server VM2 : Apache</p>
<i>Virtual machine 3</i>	<p>Name : Ubuntu Dekstop Client Operating System : Ubuntu Dekstop 20.04 (64 bit) Base Memory : 2 GB Processors : 4 Core Video Memory : 16 MB Storage : 20 GB Additional Software : Siege</p>

2) *Ubuntu Server*

Sistem operasi yang akan dipasang pada *virtual machine 1* dan 2 adalah *Ubuntu Server* versi 21.10. Sistem operasi *Ubuntu Server* yang digunakan merupakan versi 64 bit.

3) *Ubuntu Dekstop*

Pada sisi *client*, akan digunakan *virtual machine* yang memiliki sistem operasi *Ubuntu Dekstop* 64 bit versi 20.04. Pada sisi *client* akan digunakan untuk menguji kinerja *web server* yang sebelumnya sudah *terinstall* pada *virtual machine 1* dan 2.

4) *Siege*

Aplikasi yang digunakan untuk menguji performa *web server* adalah Siege versi 4.1.1. Aplikasi ini bekerja dengan cara menciptakan *virtual user* yang digunakan untuk mengakses situs *web* atau halaman *web* yang dilayani oleh *web server*.

3.4 PARAMETER HASIL PENGUJIAN

Hasil data yang didapatkan dari pengujian *load testing* menggunakan Siege berupa nilai parameter. Berdasarkan batasan masalah, parameter yang digunakan untuk menganalisis hasil *benchmark* yaitu *response time*, *transaction rate*, *throughput*, dan *elapsed time*. *Response time* merupakan waktu rata-rata yang diperlukan untuk menanggapi setiap permintaan pengguna yang disimulasikan. *Transaction rate* adalah jumlah rata-rata transaksi yang dapat ditangani *server* per detik, singkatnya transaksi dibagi dengan *elapsed time*. Transaksi adalah jumlah hit (pengaksesan) *server*. Contohnya, 10 *virtual user* [-c10] masing-masing mengakses (*hit*) satu halaman *web* sederhana yang berisi tulisan saja dengan 10 pengulangan [-r10] sehingga total transaksi yang dilakukan menjadi 100 transaksi. *Throughput* adalah jumlah rata-rata byte yang ditransfer setiap detik dari *server* ke semua pengguna yang disimulasikan. Dan *elapsed time* adalah total waktu yang dibutuhkan untuk menyelesaikan pengujian. *Elapsed time* diukur dari awal proses melakukan pengujian hingga *virtual user* terakhir menyelesaikan transaksinya.