

BAB III METODE PENELITIAN

3.1 ALAT YANG DIGUNAKAN

Pada Penelitian ini menggunakan suatu pemodelan dan simulasi dalam menganalisis performansi *webservice Openlitespeed* dan *Nginx* dalam menangani *client Request* pada *Software defined network*. Model simulasi yang diimplementasikan dalam penelitian ini menggunakan Oracle VM *Virtualbox* yang telah diinstal dua OS CentOS 7 Minimal sebagai *server Openlitespeed* dan *Nginx* dan OS Ubuntu 22.04.2 LTS Desktop sebagai *client*, GNS3 untuk membuat topologi jaringannya serta membuat SDN dengan kontroler Onos dari fitur *Docker Container* di GNS3, dan aplikasi JMETER untuk mengambil data dan parameter yang diukur yaitu *respon time*, *Throughput*, *ram Usage* dan *cpu Usage*. Jadi, perangkat yang digunakan dalam penelitian dapat dikelompokkan menjadi 2 bagian yaitu perangkat keras (*Hardware*) dan perangkat lunak (*Software*).

3.1.1 HARDWARE

Tabel 3.1 adalah daftar perangkat keras yang digunakan dalam penelitian ini. Dalam perancangan sistem untuk penelitian ini diperlukan perangkat keras untuk mendukung implementasi sistem yang dilakukan dengan benar. Persyaratan dan data teknis perangkat yang digunakan dapat ditemukan di bagian ini *table 3.1*.

Tabel 3. 1 Spesifikasi Perangkat Keras (*Hardware*)

Nama Alat	Versi	Spesifikasi Perangkat
PC	Asrock Technology 21H2	-Processor Ryzen 3 3600 -Ram 16 GB -SSD 500GB & Hardisk 1TB

3.1.2 SOFTWARE

Selain perangkat keras (*Hardware*), dalam penelitian ini dibutuhkan juga perangkat lunak (*Software*) yang dipasang pada perangkat keras (*Hardware*) yang digunakan. Untuk kebutuhan dari *Software* yang digunakan dalam penelitian dapat dilihat pada *table 3.2*. Tabel 3. 2 Spesifikasi Perangkat Lunak (*Software*)

Yang Diperlukan	Nama <i>Software</i>
Sistem Operasi	Windows 11 Pro
<i>Software</i>	<ul style="list-style-type: none"> - <i>Apache Jmeter</i> - Oracle VM Virtual Box - GNS3
Instalasi di Virtual box	<ul style="list-style-type: none"> - Centos 7 Minimal dengan Memori 2048 MB - Ubuntu 22.04.2 LTS Dekstop dengan Memori 2048
Instalasi di CentOS	<ul style="list-style-type: none"> - <i>Nginx</i> - <i>Openlitespeed</i>
Instalasi di Ubuntu 22.04.2 LTS Dekstop	<i>Apache Jmeter</i>
Instalasi di GNS3	<i>Docker Container Onos</i>

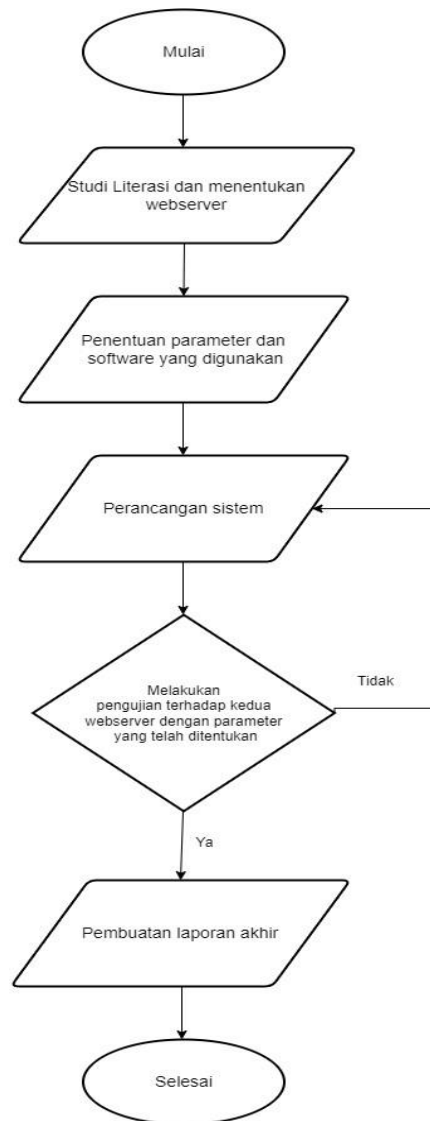
Penjelasan *Software* yang mendukung kebutuhan simulasi penelitian ini seperti yang ada pada Tabel 3.2, yaitu :

1. Oracle VM *Virtualbox* digunakan untuk tempat instalasi 2 sistem operasi virtual. CentOS 7 Minimal dengan memori 2048 MB yang digunakan sebagai *server Openlitespeed* dan *Nginx*, serta Ubuntu 22.04.2 LTS Desktop dengan memori 2048 MB yang digunakan sebagai *host*.
2. GNS3 digunakan untuk membuat topologi jaringan dengan menggunakan *controller onos*

3.2 ALUR PENELITIAN

Penelitian dilakukan dalam beberapa tahap, yaitu, tahap studi literasi, yang dimana peneliti mengkaji beberapa dari penelitian dan menentukan topik dari skripsi peneliti, dengan topik yang ditentukan mengenai *webserver*, dan peneliti menentukan *webserver* . Kemudian, peneliti menentukan parameter yang akan diuji, dan penggunaan dari *software* yang digunakan. Lalu, peneliti merancang sistem yang akan digunakan. Setelah melakukan perancangan sistem, peneliti melakukan pengujian terhadap *webserver* yang digunakan. Dengan parameter yang telah ditentukan untuk menguji bagaimana kualitas dari *webserver*

tersebut. Setelah mendapatkan hasil dari pengujian, maka peneliti membuat laporan akhir, dari penulisan awal sampai akhir.

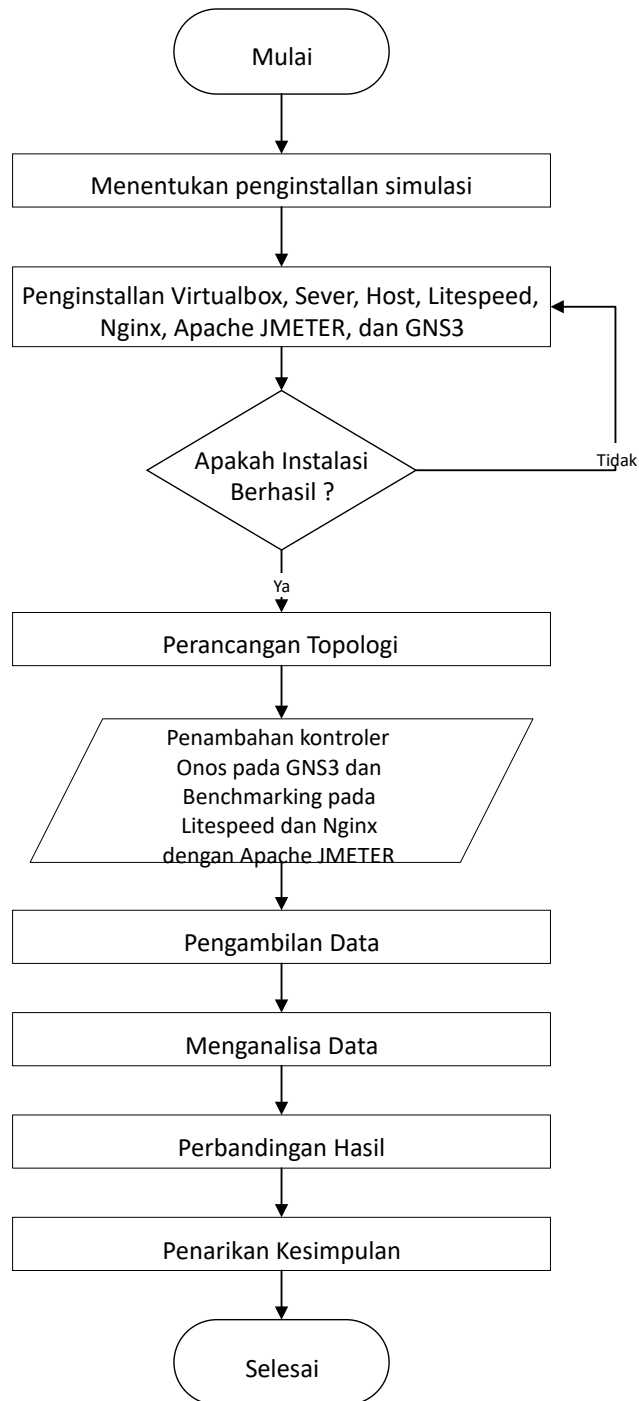


Gambar 3. 1 *Flowchart* Alur Penelitian

Seperti yang ada pada Gambar 3.1, penelitian dimulai dengan melakukan studi literasi yaitu mengumpulkan sumber pustaka dengan pembahasan mengenai *Software defined network* (SDN), *Openlitespeed*, *Nginx*, *Apache JMETER*, dan perbandingan *Openlitespeed* dengan *Nginx* dari berbagai sumber. Tujuan dari studi literasi ini untuk mendapatkan informasi mengenai penelitian sebelumnya dan mencegah penelitian yang sama. Selanjutnya melakukan perumusan masalah. Perumusan masalah bertujuan untuk mengetahui bagaimana sistem yang seharusnya dibuat dalam penelitian agar sesuai. Kemudian melakukan rancangan sistem dan

pembangunan sistem. Rancangan sistem dibuat dengan menggunakan *Openlitespeed*, *Nginx*, *Apache JMeter*, *virtualbox*, dan *GNS3*. Sistem yang akan dibangun adalah sistem yang didasarkan pada perbandingan *webservice Openlitespeed* dan *webservice Nginx* ketika menangani rangkaian *stress test* pada *Software defined network (SDN)*.

Sistem dibangun dengan membuat topologi pada *Software GNS3*. Setelah itu melakukan pengujian sistem. Pengujian sistem berfungsi memastikan bahwa sistem berjalan sesuai dengan apa yang diinginkan. Pengujian sistem ini dilakukan dengan menggunakan parameter sistem, yaitu ada 4 parameter seperti *respon time*, *Throughput*, *ram Usage* dan *cpu Usage* dalam menangani *client Request* menggunakan *tools Apache JMeter*. Setelah hasil parameter didapatkan, maka selanjutnya melakukan analisis dengan membandingkan hasil parameter dari *webservice Openlitespeed* dan *webservice Nginx*. Selanjutnya adalah tahap kesimpulan dari data yang telah dianalisis. Pada tahap kesimpulan akan menyimpulkan terhadap *webservice* mana yang terbaik ketika di *test* serangkaian *stress test*.



Gambar 3.2 *Flowchart* Alur Simulasi

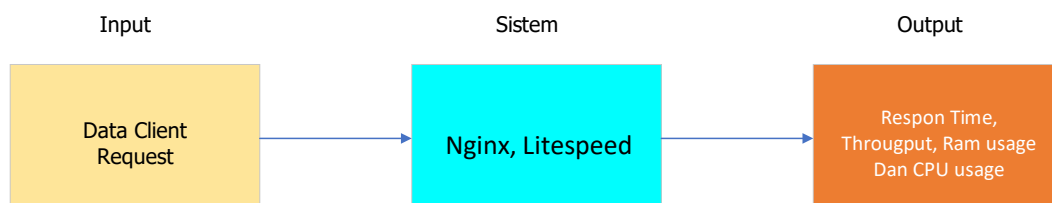
Pada Gambar 3.2 menunjukkan diagram alur simulasi. Hal pertama yang dilakukan adalah melakukan penginstalan *virtualbox*, dua CentOS 7 Minimal dengan memori 2048 MB yang menjadi *server* dan Ubuntu 22.04.2 LTS Desktop dengan memori 2048 MB yang menjadi *host* pada *Virtualbox*, *Openlitespeed* pada *server* 1, *Nginx* pada *server* 2, tool *Apache*

jmeter pada *host*, dan GNS3. Setelah penginstalan berhasil, maka langkah selanjutnya adalah membuat topologi pada GNS3. Jangan lupa tambahkan kontroler Onos secara *Container* melalui *Docker* pada GNS3 agar SDN berjalan sebagaimana mestinya.

Penelitian ini menggunakan *Docker image* Onos buatan dari Muhammad Lutfhi dengan kode muhammadluthfias/Onos pada fitur *image name* yang ditambahkan pada GNS3. *Docker image* ONOS adalah sebuah paket atau template yang berisi instalasi dari ONOS (*Open Network Operating System*) dan semua komponen pendukung yang diperlukan agar ONOS dapat berjalan di dalam lingkungan wadah (*container*) *Docker*. Jika sudah membuat topologi dan menambahkan kontroler Onos pada GNS3, maka selanjutnya tahap pengujian menggunakan *tool Apache jmeter* yang akan diinstal pada *host*. Pada tahap ini, *tool Apache jmeter* akan mengirim *Request* 500 users dengan 10 second, 500 users dengan 20 second, 1000 users dengan 10 second, 1000 users dengan 20 second, 1500 users dengan 10 second, 1500 users dengan 20 second, 2000 users dengan 10 second, 2000 users dengan 20 second ke server *Openlitespeed* dan *Nginx*. Setelah itu adalah tahap pengambilan data dengan menggunakan *tool Apache jmeter* yang nantinya *tool* ini akan mengirimkan *Request* yang sudah ditentukan agar bisa melihat *server* mana yang lebih tangguh. Kemudian selanjutnya adalah tahap perbandingan hasil antara *Openlitespeed* dan *Nginx* dalam menangani jumlah *Request* dan melihat *webserver* yang mana yang lebih unggul dalam segi performansi dan ketahanan. Dan untuk Tahap yang terakhir yaitu membuat kesimpulan dari hasil yang sudah dibandingkan.

3.3 GAMBARAN SISTEM PENELITIAN

Gambar 3.3 merupakan blok diagram perancangan. Pada penelitian ini, *server* dan *host* dibuat secara virtual menggunakan *Software Virtualbox*. Kemudian, untuk perancangan topologinya dibuat di *Software GNS3*. Pada GNS3 ini juga, kontroler Onos untuk SDN-nya akan dibuat secara kontainer. Kemudian, untuk pengujian *Request* menggunakan *tool Apache Jmeter* yang akan diinstall pada *host*. *Tools* ini akan memberikan informasi mengenai *respon time*, *Throughput*, *ram Usage* dan *cpu Usage*.

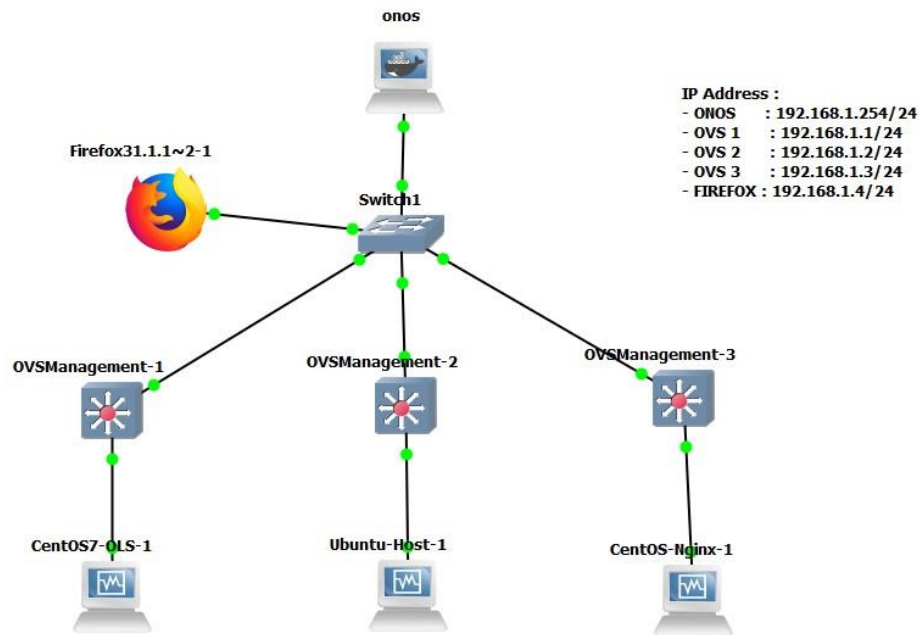


Gambar 3. 3 Blok Diagram Perancangan

Pada gambar 3.3 bisa dilihat ada 3 tahapan blok yaitu pertama ada input, proses dan output, di mana input sendiri merupakan sebuah tahapan proses inputan atau masukan di mana yang dimasukan adalah sebuah data *client Request*, di mana nantinya data *client Request* ini akan diproses dengan metode *stress test*. Metode *stress test* ini sendiri pada dasarnya untuk mengukur seberapa tahan dan seberapa kuat masing masing *webserver* ketika diberi beban, dan menentukan yang lebih unggul dengan masing masing bebannya adalah mulai dari 500 *users* dengan 10 *second*, 500 *users* dengan 20 *second*, 1000 *users* dengan 10 *second*, 1000 *users* dengan 20 *second*, 1500 *users* dengan 10 *second*, 1500 *users* dengan 20 *second*, 2000 *users* dengan 10 *second*, 2000 *users* dengan 20 *second*, Ketika masing masing *webserver* diberikan beban maka akan ada reaksi dari masing masing *webserver* tersebut, yang menghasilkan tahapan output, tahapan output sendiri disini ada *respon time*, *Throughput*, *ram Usage* dan *cpu Usage*.

3.4 PERANCANGAN TOPOLOGI PADA GNS3

Topologi pada penelitian ini dibuat dengan menggunakan *Software* GNS3. Peneliti merancang topologi sesuai dengan yang diinginkan. Kontroler Onos akan diinstal pada GNS3. *Host* akan diambil dari *Virtualbox*.



Gambar 3. 4 Perancangan Topologi Pada GNS3

Topologi pada Gambar 3.4 menunjukkan bahwa topologi menggunakan 1 kontroler Onos, 1 firefox, 3 *switch* OVS, 1 *switch* biasa, dan 3 komputer. Fungsi *switch* biasa adalah untuk menghubungkan 3 buah *switch* OVS dan 1 firefox dengan kontroler Onos. Firefox digunakan untuk mengakses Onos. Tiga komputer memiliki peran yang berbeda-beda, 1 komputer sebagai *server Openlitespeed*, 1 komputer sebagai *server Nginx*, dan 1 komputer lagi sebagai *host*, yang mana ketiga komputer tersebut akan dihubungkan dengan *virtual machine* yang akan dibuat pada *virtualbox*.

Tabel 3. 3 IP Address Perancangan Topologi Penelitian

No	Perangkat	IP
1	ONOS	192.168.1.254/24
2	OVS 1	192.168.1.1/24
3	OVS 2	192.168.1.2/24
4	OVS 3	192.168.1.3/24
5	Firefox	192.168.1.4/24
6	<i>Server Openlitespeed</i>	192.168.18.168/24
7	<i>Server Nginx</i>	192.168.18.167/24
8	<i>Host</i>	192.168.18.169/24

Pada pengujian, hanya menggunakan 1 *host* karena *host* tersebut nanti akan diinstal *Apache jmeter*, yang mana akan memberikan beban *Request* bertahap mulai dari range 500 *Request*, 1000 *Request*, 1500 *Request*, 2000 *Request* sehingga tidak perlu memberi *Request* secara manual. Untuk IP *address* perangkat-perangkat tersebut dapat dilihat pada Tabel 3.3.

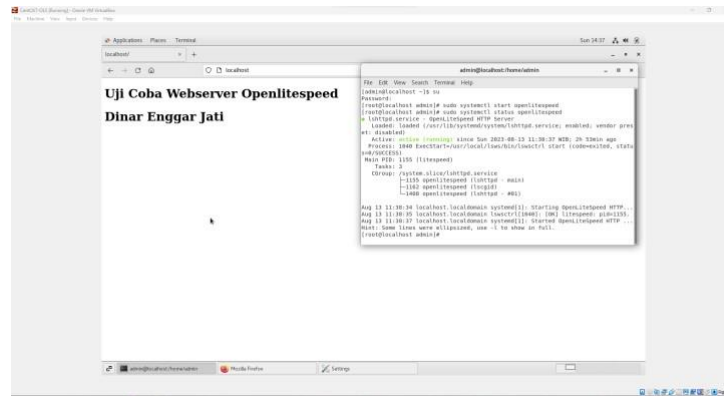
3.5 PARAMETER UJI

Analisis performansi *webserver Openlitespeed* dan *Nginx* pada *Software defined network* menggunakan parameter uji yaitu dengan melakukan pengiriman HTTP *Request* ke *webserver*. Proses ini dilakukan untuk mengetahui performansi *webserver* dalam menagani

respon time, Throughput, ram Usage dan cpu Usage. Pengujian dilakukan dengan memberi beban Request secara bertahap mulai dari 500 users dengan 20 second, sampai dengan 2000 users dengan 20 second ke masing-masing webserver Openlitespeed dan webserver Nginx.

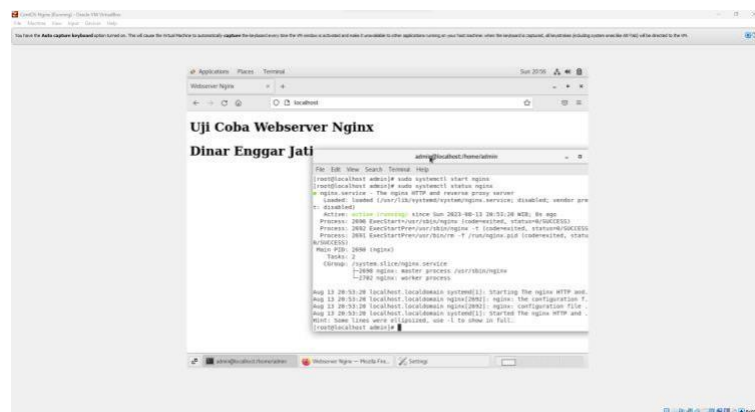
3.6 KONFIGURASI PENGUJIAN

Sebelum melakukan pengujian, peneliti terlebih dahulu melakukan konfigurasi terhadap webserver, kemudian dilakukan config pada terminal untuk install Openlitespeed dengan membuat webserver pada file html, lalu dihubungkan ke localhost / IP Address. Lalu pengujian dilakukan dengan menggunakan Apache Jmeter.



Gambar 3. 5 Konfigurasi Webserver Openlitespeed

Lalu dilakukan konfigurasi pada webserver Nginx, dengan config yang telah ditentukan, kemudian membuat file html menjadi localhost / IP Address.



Gambar 3. 6 Konfigurasi Webserver Nginx

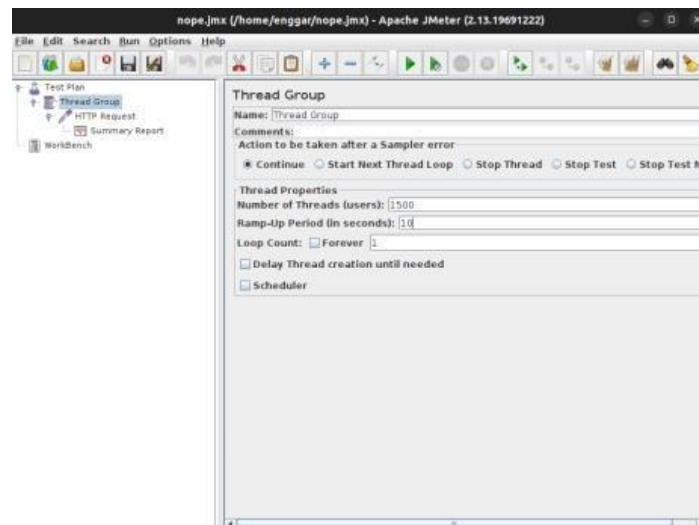
3.7 SKENARIO PENGUJIAN

Pada penelitian ini skenario pengujian yang akan diuji yaitu respon time, Throughput, ram Usage, cpu Usage. Respon time sendiri adalah waktu tanggap yang diberikan Interface ketika user me- Request permintaan ke komputer atau perbedaan waktu permintaan dikirim

dan waktu ketika telah direspon sepenuhnya, *Throughput* adalah kecepatan (rate) transfer data efektif yang diukur dalam bps. *Throughput* merupakan jumlah total kedatangan paket yang sukses dan diamati pada destination selama interval waktu tertentu. *Ram Usage* adalah penggunaan tempat penyimpanan sementara pada *server*.

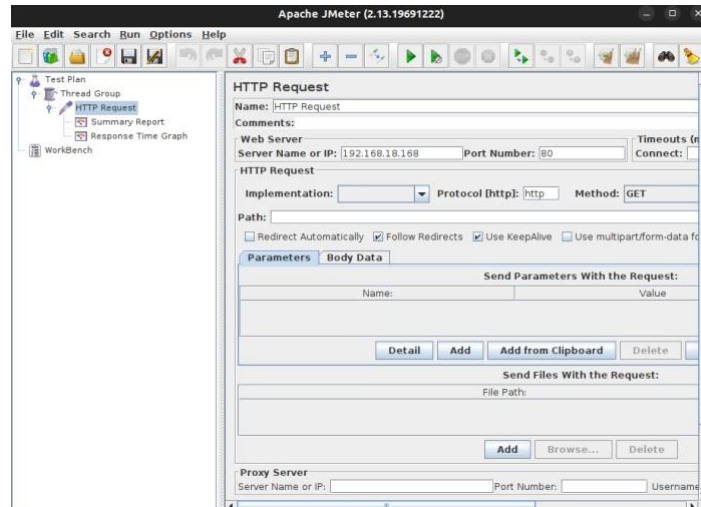
Cpu Usage adalah *core* utama untuk menerima instruksi dan melakukan tindakan berdasarkan perintah dari *user website (client)*. Keempat parameter uji tersebut untuk mengetahui bagaimana perbandingan *webserver Openlitespeed* dan *webserver Nginx* ketika menangani *client Request* pada SDN.

Pengujian dilakukan pada PC *Host* berbasis OS Ubuntu 22.04.2 LTS. Peneliti menggunakan *command* pada CLI untuk meng-*install Apache Jmeter*. Lalu peneliti melakukan konfigurasi pada *Apache Jmeter*.



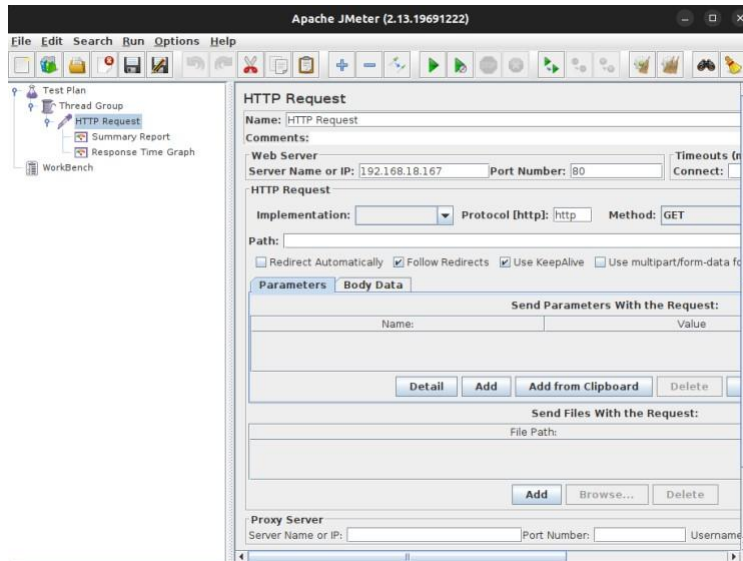
Gambar 3. 7 Tampilan Apache Jmeter

Pada gambar 3.7 dilakukan proses pengujian dilakukan dengan melakukan konfigurasi pada *Apache Jmeter*, untuk melakukan pengujian, yang di mana pengujian dilakukan dengan jumlah uji coba dengan 500 *user*, 1000, *users*, 1500 *users*, dan 2000 *users*. Adapun beberapa fitur yang digunakan untuk menguji yaitu, *Number Of Theards (users)*, digunakan untuk memasukkan jumlah user yang sudah ditentukan. Kemudian menggunakan fitur *Ramp-Up Period (in seconds)*, digunakan untuk menambahkan beban dalam satuan waktu.



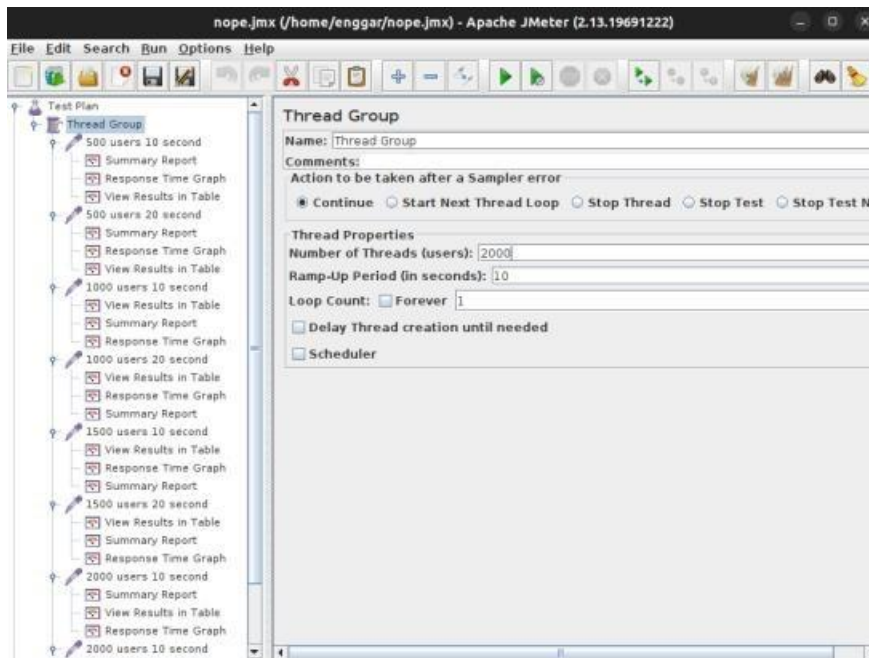
Gambar 3. 8 Konfigurasi HTTP Request Openlitespeed

Kemudian dilakukan penambahan sampler HTTP request dengan menambahkan IP address dari webstatis yang sudah terhubung kepada webserver. Pada gambar 3.6 dilakukan konfigurasi HTTP request untuk Openlitespeed. Ada beberapa fitur yang digunakan pada HTTP Request, yaitu penggunaan Server Name or IP, digunakan untuk konfigurasi IP Address yang digunakan pada webserver, kemudian ada fitur Port Number, digunakan untuk penggunaan Port yang digunakan untuk menyesuaikan port pada IP Address, port yang digunakan pada HTTP Sampler ini menggunakan port 80, dikarenakan jika pengujian web server hanya melibatkan seperti kinerja web server itu sendiri atau beban server yang tidak melibatkan keamanan maka port 80 lebih tepat, tetapi jika hanya digunakan ketika kebutuhan pengujian web servernya melibatkan keamanan server seperti enkripsi data, maka yang dibutuhkan untuk pengujian web server keamanan maka port 443 lebih tepat, karena jika pengujian keamanan menggunakan https, https dinilai lebih unggul daripada http, dikarenakan https menggunakan enkripsi data yang membuatnya semakin sulit dimanipulasi oleh pihak yang tidak berwenang, pihak berwenang disini merujuk pada individu atau kelompok seperti hacker yang tidak memiliki izin atau hak akses untuk memanipulasi data yang sedang dikirim melalui jaringan dan ada fitur terakhir yang digunakan yaitu Protocol [http], digunakan untuk memasukkan tipe sampler yang digunakan.

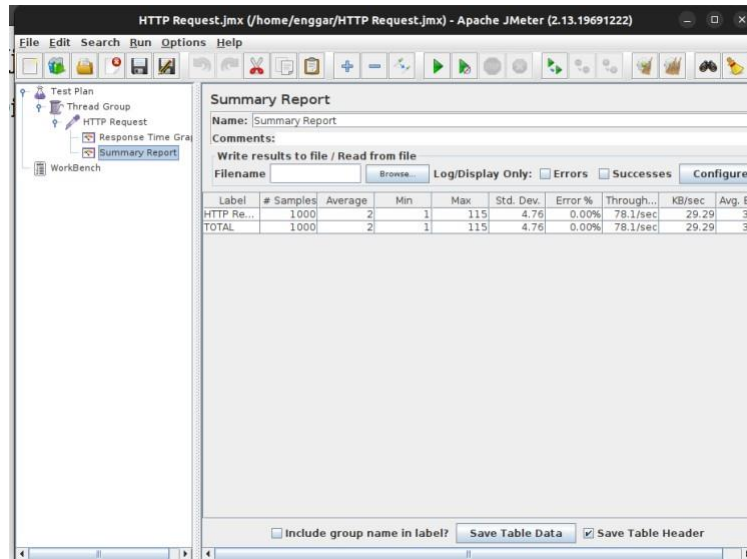


Gambar 3. 9 Konfigurasi HTTP Request Nginx

Pada gambar 3.9 dilakukan konfigurasi HTTP request dengan penambahan IP Address pada webstatis yang sudah terhubung dengan webserver Nginx.

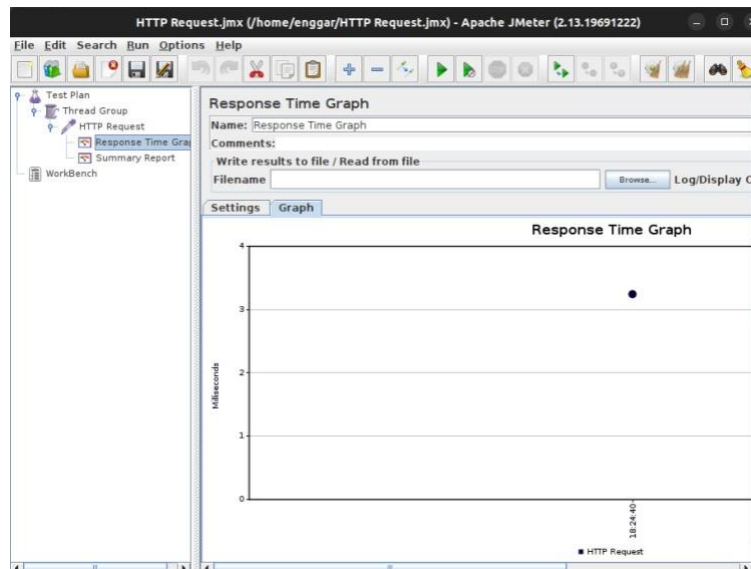


Gambar 3. 10 Konfigurasi Pengujian Apache Jmeter



Gambar 3. 11 *Interface Summary Report*

Pada Gambar 3.11 Kemudian setelah melakukan pengujian didapati hasil yang dimana hasil parameter tersebut dapat dilihat pada *Summary Report* dan *Response Time Graph* seperti pada Gambar 3. 12.



Gambar 3. 12 *Interface Respon Time Graph*

Lalu untuk setiap pengujiannya dilakukan percobaan yang berbeda, dengan user dan beban yang sudah ditentukan, seperti pada gambar 3.8.

Tabel 3.4 Percobaan Beban Request

NO	Percobaan ke-	Jumlah Beban Request

1	Percobaan 1	<i>500 users in 10 second</i>
2	Percobaan 2	<i>500 users in 20 second</i>
3	Percobaan 3	<i>1000 users in 10 second</i>
4	Percobaan 4	<i>1000 users in 20 second</i>
5	Percobaan 5	<i>1500 users in 10 second</i>
6	Percobaan 6	<i>1500 users in 20 second</i>
7	Percobaan 7	<i>2000 users in 10 second</i>
8	Percobaan 8	<i>2000 users in 20 second</i>