

Evaluasi Kinerja Web Server Nginx Dan Apache2 Pada Sistem Debian Menggunakan Metode Stress Testing

Angela Wita Simanullang¹, Erich Ricardo², Lotar Mateus Sinaga³

Universitas Katolik Santo Thomas, Medan, Indonesia^{1,2,3}

angelawita38@gmail.com¹, ericmarbun3@gmail.com², lotarmateus88@gmail.com³

*Corresponding author : ²ericmarbun3@gmail.com²

Abstrak—Meningkatnya kebutuhan terhadap layanan berbasis web yang cepat dan stabil menjadikan pemilihan web server sebagai faktor penting dalam implementasi sistem. Penelitian ini bertujuan mengevaluasi dan membandingkan kinerja Apache2 dan Nginx pada sistem operasi Debian 12 menggunakan metode *stress testing*. Pengujian dilakukan pada lingkungan implementasi yang identik menggunakan Apache Benchmark (*ab*) dengan 5.000 *request* dan 100 *concurrent connection*. Parameter yang dianalisis meliputi *Requests per Second*, *Time per Request*, *Transfer Rate*, dan *Failed Requests*. Hasil penelitian menunjukkan bahwa Nginx memperoleh 5500.65 *Requests per Second*, *Time per Request* sebesar 18.180 ms, dan *Transfer Rate* sebesar 4555.23 KBytes/sec, sedangkan Apache2 memperoleh 439.66 *Requests per Second*, *Time per Request* sebesar 227.448 ms, dan *Transfer Rate* sebesar 2607.48 KBytes/sec. Kedua web server berhasil menyelesaikan seluruh *request* tanpa *Failed Requests*. Penelitian ini memberikan evaluasi objektif terhadap performa kedua web server sehingga dapat menjadi referensi dalam pemilihan web server sesuai kebutuhan implementasi layanan berbasis web.

Abstract— *The increasing demand for fast and reliable web-based services makes web server selection an important factor in system implementation. This study aims to evaluate and compare the performance of Apache2 and Nginx on the Debian 12 operating system using a stress testing approach. Both web servers were deployed in an identical environment and evaluated using Apache Benchmark (ab) with 5,000 requests and 100 concurrent connections. The evaluated parameters included Requests per Second, Time per Request, Transfer Rate, and Failed Requests. The results showed that Nginx achieved 5500.65 Requests per Second, a Time per Request of 18.180 ms, and a Transfer Rate of 4555.23 KBytes/sec, while Apache2 achieved 439.66 Requests per Second, a Time per Request of 227.448 ms, and a Transfer Rate of 2607.48 KBytes/sec. Both web servers completed all requests without Failed Requests. These findings provide an objective performance evaluation and can serve as a reference for selecting web servers based on implementation requirements.*

Keywords: Web Server, Apache2, Nginx, Debian 12, Stress Testing, Apache Benchmark.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-nc/4.0/) license.



1. Pendahuluan

Perkembangan teknologi informasi dan komunikasi telah mendorong pemanfaatan layanan berbasis web pada berbagai sektor, seperti pendidikan, pemerintahan, kesehatan, bisnis, hingga layanan publik. Seiring dengan meningkatnya jumlah pengguna internet dan transformasi digital, kebutuhan terhadap sistem yang mampu menyediakan layanan secara cepat, stabil, dan berkelanjutan semakin tinggi. Web server merupakan salah satu komponen utama yang berfungsi menerima permintaan (*request*) dari pengguna melalui protokol HTTP/HTTPS, kemudian memproses dan mengirimkan respons sesuai dengan layanan yang diminta. Kinerja web server berpengaruh terhadap kualitas layanan karena berkaitan dengan kecepatan respons, kemampuan melayani banyak permintaan secara bersamaan, serta efisiensi penggunaan sumber daya server [1]. Oleh karena itu, pemilihan web server yang tepat menjadi salah satu faktor penting dalam menjaga ketersediaan layanan dan meningkatkan kualitas pengalaman pengguna.

Apache HTTP Server dan Nginx merupakan dua web server yang paling banyak digunakan dalam implementasi berbagai layanan berbasis web. Apache dikenal memiliki fleksibilitas yang tinggi melalui dukungan modul yang lengkap serta kompatibilitas terhadap berbagai teknologi web. Di sisi lain, Nginx dikembangkan menggunakan arsitektur *event-driven* yang mampu menangani koneksi dalam jumlah besar dengan penggunaan sumber daya yang lebih efisien. Perbedaan karakteristik tersebut menyebabkan kedua web server memiliki keunggulan masing-masing sesuai kebutuhan implementasi sistem, sehingga diperlukan pengujian untuk mengetahui performa sebenarnya ketika melayani permintaan dari pengguna [2]. Hasil evaluasi tersebut dapat menjadi dasar bagi administrator sistem dalam menentukan web server yang sesuai dengan kebutuhan implementasi.

Evaluasi performa web server umumnya dilakukan menggunakan metode *stress testing* atau *benchmarking* untuk mengetahui kemampuan server dalam menangani sejumlah permintaan secara bersamaan. Pengujian dilakukan dengan memberikan beban tertentu sehingga dapat diketahui karakteristik performa masing-masing web server berdasarkan parameter seperti *Requests per Second*, *Time per Request*, *Transfer Rate*, maupun *Failed Requests*. Hasil pengujian tersebut menjadi dasar dalam menilai efektivitas web server ketika digunakan pada kondisi lalu lintas yang tinggi [3]. Melalui proses evaluasi tersebut, administrator sistem dapat menentukan solusi yang paling sesuai berdasarkan kebutuhan layanan tanpa hanya bergantung pada popularitas suatu web server.

Penelitian terbaru menunjukkan bahwa evaluasi performa web server tidak hanya dilakukan berdasarkan jumlah permintaan yang berhasil diproses, tetapi juga mempertimbangkan parameter seperti *response time*, *throughput*, dan *latency* untuk menilai kualitas layanan pada berbagai tingkat beban kerja. Penggunaan parameter tersebut memberikan gambaran yang lebih komprehensif mengenai kemampuan web server dalam menangani peningkatan jumlah permintaan pengguna [4].

Berbagai penelitian mengenai perbandingan performa web server telah dilakukan menggunakan metode dan lingkungan pengujian yang berbeda. Salah satu penelitian membandingkan Apache, Nginx, dan Lighttpd pada sistem Debian menggunakan perangkat lunak WRK dengan variasi jumlah pengguna dan durasi pengujian. Hasil penelitian tersebut menunjukkan bahwa setiap web server memiliki karakteristik performa yang berbeda bergantung pada arsitektur, konfigurasi, serta kondisi beban yang diberikan. Penelitian tersebut menunjukkan bahwa setiap web server memiliki karakteristik performa yang berbeda bergantung pada arsitektur, konfigurasi, serta kondisi beban yang diberikan. Oleh karena itu, proses *benchmarking* menjadi langkah penting untuk memperoleh gambaran objektif mengenai kemampuan masing-masing web server sebelum diterapkan pada lingkungan operasional [5].

Selain penelitian yang berfokus pada pengujian performa, beberapa penelitian juga mengembangkan implementasi web server pada lingkungan yang lebih kompleks, seperti penggunaan Docker dan Kubernetes untuk mendukung mekanisme *failover* serta *autoscaling*. Pendekatan tersebut bertujuan meningkatkan ketersediaan layanan dan skalabilitas sistem melalui pengelolaan beberapa node secara otomatis. Meskipun demikian, penelitian tersebut lebih menitikberatkan pada aspek keandalan infrastruktur dibandingkan evaluasi performa web server secara langsung menggunakan metode *stress testing* [6]. Selain itu, penggunaan lingkungan pengujian, perangkat lunak *benchmark*, maupun parameter pengujian yang berbeda pada setiap penelitian menyebabkan hasil evaluasi sulit dibandingkan secara langsung sehingga masih diperlukan penelitian pada lingkungan yang lebih seragam.

Berdasarkan telaah terhadap penelitian terdahulu, masih terdapat kesenjangan penelitian (*research gap*) terkait evaluasi performa Apache2 dan Nginx pada sistem operasi Debian 12 menggunakan lingkungan implementasi yang identik serta metode pengujian yang sama. Sebagian besar penelitian sebelumnya menggunakan sistem operasi, perangkat lunak *benchmark*, maupun infrastruktur yang berbeda sehingga hasil pengujian belum dapat dibandingkan secara objektif. Oleh karena itu, **kebaruan (novelty)** penelitian ini terletak pada evaluasi dan perbandingan performa Apache2 dan Nginx menggunakan Apache Benchmark (*ab*) pada sistem operasi Debian 12 dengan lingkungan pengujian yang identik. Penelitian ini mengevaluasi kedua web server berdasarkan parameter *Requests per Second*, *Time per Request*, *Transfer Rate*, dan *Failed Requests* sehingga mampu memberikan gambaran yang lebih objektif mengenai karakteristik performa masing-masing web server.

Berdasarkan *research gap* dan *novelty* tersebut, penelitian ini bertujuan mengevaluasi serta membandingkan kinerja Apache2 dan Nginx pada sistem operasi Debian 12 menggunakan metode *stress testing* dengan Apache Benchmark (*ab*). Hasil penelitian diharapkan dapat memberikan informasi mengenai karakteristik performa kedua web server serta menjadi referensi bagi administrator sistem dalam menentukan web server yang sesuai dengan kebutuhan implementasi layanan berbasis web.

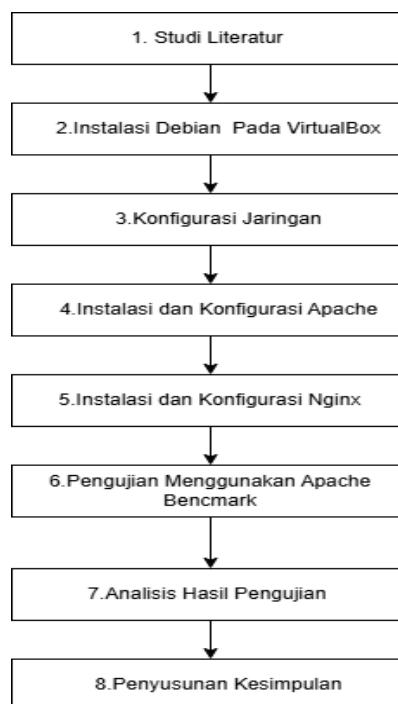
2. Metodologi Penelitian

Penelitian ini menggunakan metode eksperimen untuk mengevaluasi dan membandingkan kinerja web server Apache2 dan Nginx pada sistem operasi Debian 12 melalui pengujian *stress testing*. Pendekatan eksperimen dipilih karena memungkinkan proses pengujian dilakukan pada kondisi yang sama sehingga hasil yang diperoleh dapat dibandingkan secara objektif. Pengujian difokuskan pada pengukuran parameter *Requests per Second*, *Time per Request*, *Transfer Rate*, dan *Failed Requests* sebagai indikator utama dalam menilai performa masing-masing web server [7]. Penelitian dilaksanakan pada lingkungan pengujian yang identik sehingga setiap web server memperoleh perlakuan yang sama selama proses pengujian berlangsung.

Tabel 1. Spesifikasi Perangkat Penelitian

Komponen	Spesifikasi
Perangkat Penelitian	Laptop ASUS ExpertBook B1402CBA
Processor	12th Gen Intel® Core™ i5-1235U
Jumlah Core	10 Core
Logical Processor	12 CPU
Memori (RAM)	16 GB (16384 MB)
Media Penyimpanan	SSD 512 GB
Virtual Machine	Oracle VirtualBox
Sistem Operasi Guest	Debian GNU/Linux 12
Web Server	Apache2 2.4.57 dan Nginx 1.22.1
Sistem Operasi Host	Windows 11 Pro 64-bit
Perangkat Benchmark	Apache Benchmark (ab)

Tabel 1 menunjukkan spesifikasi perangkat keras dan perangkat lunak yang digunakan selama penelitian. Seluruh proses instalasi, konfigurasi, dan pengujian dilakukan menggunakan lingkungan perangkat yang sama sehingga hasil pengujian dapat direplikasi dan dibandingkan secara objektif dengan penelitian lain. Tahapan penelitian dimulai dengan melakukan studi literatur, kemudian dilanjutkan dengan instalasi sistem operasi Debian 12 pada lingkungan virtual menggunakan Oracle VirtualBox. Setelah sistem berhasil dijalankan, dilakukan instalasi dan konfigurasi Apache2 serta Nginx secara terpisah agar kedua web server dapat diuji pada kondisi yang sama. Tahap berikutnya adalah pelaksanaan stress testing menggunakan Apache Benchmark (ab) dengan parameter yang telah ditentukan. Seluruh data hasil pengujian kemudian dianalisis untuk memperoleh perbandingan performa kedua web server berdasarkan parameter yang diamati. Tahapan penelitian yang dilakukan pada penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Berdasarkan Gambar 1, penelitian dilakukan secara bertahap mulai dari studi literatur hingga analisis hasil pengujian. Setiap tahapan saling berkaitan sehingga proses implementasi, pengujian, dan analisis dapat dilakukan secara sistematis dan menghasilkan data yang dapat dibandingkan secara objektif.

Studi literatur dilakukan dengan mengumpulkan berbagai referensi yang berkaitan dengan implementasi web server, evaluasi performa, sistem operasi Debian, serta metode pengujian yang digunakan. Referensi tersebut menjadi dasar dalam menentukan tahapan penelitian, parameter pengujian, serta teknik analisis yang diterapkan sehingga penelitian dapat dilaksanakan secara sistematis dan sesuai dengan tujuan yang telah ditetapkan [8].

Tahap implementasi diawali dengan instalasi sistem operasi Debian 12 pada Oracle VirtualBox sebagai lingkungan server virtual. Setelah proses instalasi selesai, dilakukan konfigurasi jaringan agar server dapat diakses melalui jaringan lokal. Selanjutnya Apache2 dan Nginx diinstal secara bergantian pada lingkungan sistem yang sama hingga layanan HTTP dapat diakses menggunakan browser client melalui alamat IP server. Seluruh proses konfigurasi dilakukan dengan prosedur yang identik untuk memastikan bahwa hasil pengujian diperoleh pada kondisi yang setara. Pengujian performa dilakukan menggunakan Apache Benchmark (ab) dengan parameter sebagai berikut.

ab -n 5000 -c 100 http://192.168.1.46/

Keterangan:

- -n 5000 : jumlah request yang dikirimkan ke web server.
- -c 100 : jumlah koneksi simultan (concurrent connection).

Parameter pengujian sebesar 5000 *request* dan 100 *concurrent connection* dipilih karena mampu merepresentasikan kondisi beban akses menengah hingga tinggi pada web server. Parameter tersebut dinilai cukup untuk mengevaluasi kemampuan server dalam menangani banyak permintaan secara bersamaan tanpa menyebabkan kegagalan layanan selama proses pengujian. Selain itu, konfigurasi tersebut juga banyak digunakan pada penelitian *benchmarking* web server sehingga hasil penelitian dapat dibandingkan dengan penelitian sebelumnya.

Parameter yang diamati pada penelitian ini meliputi:

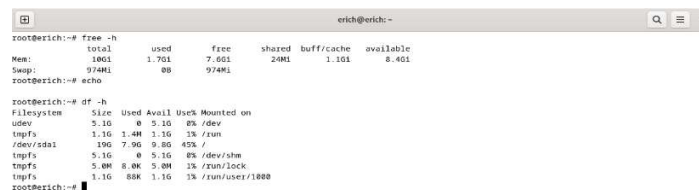
1. Requests per Second
2. Time per Request
3. Transfer Rate
4. Failed Requests

Seluruh parameter tersebut digunakan sebagai dasar dalam mengevaluasi performa Apache2 dan Nginx selama menerima beban akses yang sama.

Data hasil pengujian Apache2 dan Nginx dianalisis menggunakan metode komparatif dengan membandingkan nilai Requests per Second, Time per Request, Transfer Rate, dan Failed Requests yang diperoleh dari Apache Benchmark. Web server yang memiliki nilai Requests per Second dan Transfer Rate lebih tinggi serta Time per Request lebih rendah dinilai memiliki performa yang lebih baik. Metode komparatif digunakan karena mampu memberikan gambaran objektif mengenai perbedaan performa kedua web server berdasarkan parameter yang sama sehingga hasil evaluasi dapat dijadikan dasar dalam menyimpulkan kelebihan dan kekurangan Apache2 maupun Nginx pada sistem operasi Debian 12.

3. Hasil dan Pembahasan

Sebelum implementasi web server dilakukan, spesifikasi perangkat server terlebih dahulu diperiksa untuk memastikan kapasitas memori dan media penyimpanan mencukupi dalam mendukung instalasi Apache2, Nginx, serta pelaksanaan pengujian menggunakan Apache Benchmark (ab).



```

erich@erich:~$ free -h
              total        used        free      shared  buff/cache   available
Mem:           186G         1.7G         7.6G          20M         1.1G         8.4G
Swap:          0G          0B          0G

erich@erich:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            5.1G     0  5.1G   0% /dev
tmpfs           1.1G   1.1G   0B 100% /run
/dev/sda1       19G   7.9G   9.8G  45% /
tmpfs           5.1G     0  5.1G   0% /dev/shm
tmpfs           5.0M   8.0K   5.0M   1% /run/lock
tmpfs           1.1G   88K   1.1G   1% /run/user/1000
  
```

Gambar 2. Informasi kapasitas memori dan media penyimpanan pada server Debian 12 menggunakan perintah *free -h* dan *df -h*.

Berdasarkan Gambar 2, server Debian 12 memiliki kapasitas memori dan ruang penyimpanan yang memadai untuk menjalankan layanan web server. Informasi tersebut menjadi dasar bahwa lingkungan pengujian telah siap digunakan pada proses implementasi dan pengujian performa Apache maupun Nginx.

Sebelum layanan web server diakses oleh client, dilakukan pemeriksaan konfigurasi alamat IP server menggunakan perintah `ip a`. Alamat IP ini digunakan sebagai media komunikasi antara server dan komputer client pada jaringan lokal.

```
erich@erich:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:00:53:ac brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.46/24 brd 192.168.1.255 scope global dynamic ens3
        valid_lft 86758sec preferred_lft 86758sec
    inet6 2407:0:3003:012a::400:27ff:febd:53ac/64 scope global dynamic mngtmpaddr
        valid_lft 1403sec preferred_lft 1403sec
    inet6 fe80::a0e:27ff:febd:53ac/64 scope link
        valid_lft forever preferred_lft forever
erich@erich:~$
```

Gambar 3. Konfigurasi alamat IP server Debian 12 menggunakan perintah `ip a`.

Berdasarkan Gambar 3, server telah memiliki alamat IP yang digunakan sebagai alamat akses dari komputer client. Konfigurasi jaringan yang benar memastikan proses implementasi serta pengujian web server dapat dilakukan tanpa kendala komunikasi antar perangkat. Implementasi Apache dilakukan dengan memastikan layanan Apache telah berjalan pada sistem operasi Debian 12. Status layanan diperiksa menggunakan perintah `systemctl status apache2`, kemudian dilakukan pengujian akses melalui browser client.

```
erich@erich:~$ sudo systemctl status apache2
* apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-06-25 09:21:02 WIB; 2mn 38s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 4508 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 4512 (apache2)
     Tasks: 6 (limit: 12346)
    Memory: 17.3M
         CPU: 204ms
   CGroup: /system.slice/apache2.service
           └─4512 /usr/sbin/apache2 -k start
             └─4513 /usr/sbin/apache2 -k start
               └─4514 /usr/sbin/apache2 -k start
                 └─4515 /usr/sbin/apache2 -k start
                   └─4516 /usr/sbin/apache2 -k start
                     └─4517 /usr/sbin/apache2 -k start

Jun 25 09:21:02 erich systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jun 25 09:21:02 erich systemd[1]: Started apache2.service - The Apache HTTP Server.
erich@erich:~$
```

Gambar 4. Status layanan Apache Web Server pada Debian 12.

Hasil pada Gambar 4 menunjukkan bahwa layanan Apache berada pada status active (running), yang menandakan bahwa web server telah berhasil dijalankan dan siap menerima permintaan dari client.

```
File Machine View Input Device Help
Server Software: Apache/2.4.67
Server Hostname: 192.168.1.46
Server Port: 80
Document Path: /
Document Length: 5585 bytes
Concurrency Level: 100
Time taken for tests: 11.372 seconds
Complete requests: 5000
Failed requests: 0
Total transferred: 30365000 bytes
HTML transferred: 27925000 bytes
Requests per second: 439.66 [#/sec] (mean)
Time per request: 227.448 [ms] (mean)
Time per request: 2.274 [ms] (mean, across all concurrent requests)
Transfer rate: 2607.48 [Kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:  0    0  1.0-0    0    9
Processing: 18  225  88.9-220  673
Waiting:  9  199  85.5-196  641
Total:  22  225  88.8-221  674

Percentage of the requests served within a certain time (ms)
 50%    221
 66%    259
 75%    279
 80%    296
 90%    335
```

Gambar 5. Hasil benchmark performa Apache menggunakan Apache Benchmark (ab).

Berdasarkan hasil benchmark pada Gambar 5, Apache2 memperoleh nilai Requests per Second sebesar 439,66 requests/sec. Nilai tersebut menunjukkan bahwa Apache2 mampu melayani permintaan pengguna dengan baik, namun peningkatan jumlah koneksi simultan menyebabkan waktu respons meningkat. Kondisi tersebut dipengaruhi oleh arsitektur process-driven yang digunakan Apache2, di mana setiap permintaan diproses menggunakan proses atau thread tersendiri. Ketika jumlah koneksi meningkat, penggunaan sumber daya sistem juga bertambah sehingga memengaruhi performa web server [9].

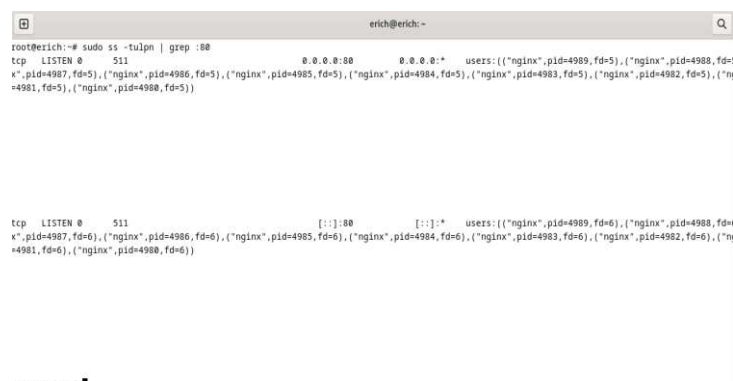


```
erich@erich:~$ sudo systemctl status nginx
nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-06-25 09:58:02 WIB; 1min 49s ago
     Docs: man:nginx(8)
   Process: 4978 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 4979 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 4980 (nginx)
    Tasks: 10 (limit: 12345)
   Memory: 6.9M
      CPU: 58ms
   CGroup: /system.slice/nginx.service
           └─4980 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─4981 "nginx: worker process"
               └─4982 "nginx: worker process"
                 └─4983 "nginx: worker process"
                   └─4984 "nginx: worker process"
                     └─4985 "nginx: worker process"
                       └─4986 "nginx: worker process"
                         └─4987 "nginx: worker process"
                           └─4988 "nginx: worker process"
                             └─4989 "nginx: worker process"

Jun 25 09:58:02 erich systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Jun 25 09:58:02 erich systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
erich@erich:~$
```

Gambar 6. Status layanan Nginx Web Server pada Debian 12.

Hasil pada Gambar 6 menunjukkan bahwa layanan Nginx berada pada status active (running) sehingga siap digunakan sebagai web server.



```
erich@erich:~$ sudo ss -tlnp | grep :80
tcp LISTEN 0 511 0.0.0.0:* 0.0.0.0:* users:(("nginx",pid=4989,fd=5),("nginx",pid=4988,fd=4),("nginx",pid=4987,fd=5),("nginx",pid=4986,fd=5),("nginx",pid=4985,fd=5),("nginx",pid=4984,fd=5),("nginx",pid=4983,fd=5),("nginx",pid=4982,fd=5),("nginx",pid=4981,fd=5),("nginx",pid=4980,fd=5))

tcp LISTEN 0 511 [::]:80 [::]:* users:(("nginx",pid=4989,fd=6),("nginx",pid=4988,fd=4),("nginx",pid=4987,fd=6),("nginx",pid=4986,fd=6),("nginx",pid=4985,fd=6),("nginx",pid=4984,fd=6),("nginx",pid=4983,fd=6),("nginx",pid=4982,fd=6),("nginx",pid=4981,fd=6),("nginx",pid=4980,fd=6))

erich@erich:~$
```

Gambar 7. Pemeriksaan penggunaan port 80 oleh layanan Nginx.

Berdasarkan Gambar 8, layanan Nginx telah menggunakan port 80 (HTTP) sebagai port komunikasi web server, sehingga implementasi Nginx berhasil dilakukan dengan baik.

```
Document Path: /
Document Length: 615 bytes

Concurrency Level: 100
Time taken for tests: 0.909 seconds
Complete requests: 5000
Failed requests: 0
Total transferred: 4240000 bytes
HTML transferred: 3075000 bytes
Requests per second: 5500.65 [#/sec] (mean)
Time per request: 18.180 [ms] (mean)
Time per request: 0.182 [ms] (mean, across all concurrent requests)
Transfer rate: 4555.23 [Kbytes/sec] received
```

Connection Times (ms)				
	min	mean[+/-sd]	median	max
Connect:	0	8 2.7	8	25
Processing:	2	10 3.6	10	31
Waiting:	1	7 3.4	7	29
Total:	8	18 4.6	17	41

```
Percentage of the requests served within a certain time (ms)
50% 17
66% 18
75% 19
80% 20
90% 23
95% 29
98% 31
99% 34
```

Gambar 9. Hasil benchmark performa Nginx menggunakan Apache Benchmark (ab).

Berdasarkan hasil benchmark pada Gambar 9, Nginx memperoleh nilai **Requests per Second** sebesar **5500,65 requests/sec** dengan **Time per Request** sebesar **18,180 ms**. Hasil tersebut menunjukkan bahwa Nginx memiliki kemampuan menangani permintaan dalam jumlah besar dengan waktu respons yang lebih rendah dibandingkan Apache2. Keunggulan tersebut dipengaruhi oleh penerapan arsitektur *event-driven* yang memungkinkan satu *worker process* menangani banyak koneksi secara bersamaan melalui mekanisme *asynchronous I/O*, sehingga penggunaan CPU dan memori menjadi lebih efisien [10]. Kondisi tersebut menyebabkan Nginx mampu mempertahankan performa yang lebih stabil ketika menerima jumlah koneksi simultan yang tinggi dibandingkan Apache2.

Untuk mempermudah proses analisis, hasil benchmark Apache2 dan Nginx menggunakan Apache Benchmark (ab) dirangkum dalam Tabel 2. Penyajian hasil dalam bentuk tabel bertujuan untuk memperlihatkan perbedaan performa kedua web server berdasarkan parameter pengujian yang sama.

Tabel 2. Perbandingan Hasil Pengujian Apache2 dan Nginx Menggunakan Apache Benchmark (ab)

Parameter	Apache2	Nginx	Keterangan
Web Server	Apache2 2.4.57 (Debian)	Nginx 1.22.1 (Debian)	Versi web server yang digunakan
Concurrency Level	100	100	Jumlah koneksi simultan
Complete Requests	5000	5000	Jumlah <i>request</i> yang berhasil diproses
Failed Requests	0	0	Tidak terdapat kegagalan selama pengujian
Time Taken for Tests	11.372 s	0.909 s	Semakin kecil semakin baik
Requests per Second	439.66 req/sec	5500.65 req/sec	Semakin besar semakin baik
Time per Request (mean)	227.448 ms	18.180 ms	Semakin kecil semakin baik
Time per Request (all concurrent)	2.274 ms	0.182 ms	Waktu rata-rata pada seluruh koneksi
Transfer Rate	2607.48 KB/sec	4555.23 KB/sec	Semakin besar semakin baik

Sumber: Hasil pengujian menggunakan Apache Benchmark (ab).

Berdasarkan Tabel 2, Nginx menunjukkan performa yang lebih baik dibandingkan Apache2 pada seluruh parameter utama pengujian. Nilai Requests per Second dan Transfer Rate yang lebih tinggi menunjukkan bahwa Nginx mampu memproses permintaan pengguna dengan kapasitas yang lebih besar, sedangkan nilai Time Taken for Tests dan Time per Request yang lebih rendah menunjukkan bahwa Nginx memberikan waktu respons yang lebih cepat dibandingkan Apache2. Selain itu, kedua web server berhasil menyelesaikan seluruh proses pengujian tanpa Failed Requests, sehingga keduanya memiliki tingkat keandalan yang baik pada lingkungan implementasi yang digunakan. Temuan ini sejalan dengan penelitian sebelumnya yang menyatakan bahwa Nginx memiliki kemampuan lebih baik dalam menangani beban akses tinggi dibandingkan Apache[11].

Perbedaan performa tersebut dipengaruhi oleh karakteristik arsitektur yang digunakan masing-masing web server. Nginx menerapkan arsitektur *event-driven* yang memungkinkan satu *worker process* menangani banyak koneksi secara bersamaan melalui mekanisme *asynchronous I/O*, sehingga penggunaan CPU dan memori menjadi lebih efisien. Sebaliknya, Apache2 menggunakan pendekatan *process-driven* yang memerlukan proses atau *thread* tersendiri untuk setiap koneksi sehingga kebutuhan sumber daya meningkat ketika jumlah koneksi simultan bertambah. Kondisi tersebut menyebabkan Nginx mampu mempertahankan performa yang lebih stabil pada beban akses yang tinggi dibandingkan Apache2[12].

Hasil penelitian ini menunjukkan bahwa pemilihan web server perlu disesuaikan dengan karakteristik aplikasi dan kebutuhan implementasi. Nginx lebih sesuai digunakan pada layanan dengan lalu lintas tinggi yang membutuhkan waktu respons rendah dan *throughput* tinggi, sedangkan Apache2 masih menjadi pilihan yang baik untuk aplikasi web dengan tingkat akses rendah hingga menengah karena memiliki fleksibilitas konfigurasi dan dukungan modul yang lebih lengkap. Temuan ini memperkuat hasil penelitian sebelumnya mengenai pentingnya pemilihan web server berdasarkan kebutuhan implementasi agar diperoleh performa sistem yang optimal[13].

4. Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa Nginx memiliki performa yang lebih baik dibandingkan Apache2 pada sistem operasi Debian 12 berdasarkan parameter Requests per Second, Time

per Request, dan Transfer Rate yang diperoleh melalui pengujian menggunakan Apache Benchmark (ab). Nginx mampu menangani beban akses yang sama dengan waktu respons lebih rendah serta tingkat transfer data yang lebih tinggi dibandingkan Apache2, sementara kedua web server sama-sama berhasil menyelesaikan seluruh pengujian tanpa Failed Requests. Hasil penelitian ini menunjukkan bahwa penerapan arsitektur *event-driven* pada Nginx memberikan efisiensi yang lebih baik dalam menangani koneksi simultan dibandingkan pendekatan *process-driven* pada Apache2. Dengan demikian, penelitian ini memberikan kontribusi berupa evaluasi objektif terhadap performa kedua web server pada lingkungan implementasi yang identik sehingga dapat menjadi referensi dalam pemilihan web server sesuai kebutuhan layanan berbasis web. Penelitian ini masih terbatas pada satu skenario pengujian menggunakan 5.000 request dan 100 concurrent connection, sehingga penelitian selanjutnya disarankan menggunakan variasi beban akses, perangkat *benchmark* yang berbeda, serta mengevaluasi penggunaan sumber daya server agar diperoleh hasil yang lebih komprehensif.

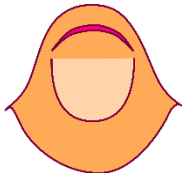
5. Daftar Pustaka

- [1] D. Kunda, "Web Server Performance of Apache and Nginx : A Systematic Literature Review," vol. 8, no. 2, pp. 43–52, 2017.
- [2] A. Y. Chandra, "Analisis Performansi Antara Apache & Nginx Web Server dalam Menangani Client Request," pp. 48–56, 2019, doi: 10.30864/jsi.v14i1.248.
- [3] D. Teknik, S. Vokasi, and U. G. Mada, "Analisis Kinerja Web Server Apache , Nginx , dan Caddy dengan Metode Stress Testing Menggunakan Autocannon," vol. 6, no. 2, pp. 96–100, 2025.
- [4] K. Prasojo and A. M. F. Fadhlurrahman, "Analisis Performa Load balancing pada Web Server Menggunakan Nginx," vol. 3, no. 1, pp. 30–39, 2025.
- [5] A. M. Herman, W. Wicoksono, P. A. Panchadria, N. Linda, and B. Laela, "Performance Comparison of NGINX , Apache , and Lighttpd Using WRK on a Debian," vol. 8, no. 1, 2025, doi: 10.32877/bt.v8i1.2661.
- [6] Y. T. Sumbogo, M. Data, and R. A. Siregar, "Implementasi Failover Dan Autoscaling Kontainer Web Server Nginx Pada Docker Menggunakan Kubernetes," vol. 2, no. 12, pp. 6849–6854, 2018.
- [7] K. Pengantar, "ANALISA PERBANDINGAN KINERJA WEB SERVER APACHE , NGINX , DAN LITESPEED DENGAN MENGGUNAKAN METODE STRESS TEST UNIVERSITAS ISLAM RIAU," 2021.
- [8] M. F. Raharjo, "Evaluasi Kinerja Web Server Apache menggunakan Protokol HTTP2," vol. 2, no. 1, pp. 19–31, doi: 10.36079/lamintang.jetas-0201.92.
- [9] V. N. Nguyen, "Comparative Performance Evaluation of Web Servers," vol. 31, no. 3, pp. 28–34, 2017.
- [10] M. Pranata, A. Wijayanto, and M. F. Sidiq, "Performance Comparison of Web Server Application on Single Board Computer".
- [11] H. Tang and L. Thompson, "Analysis of HTTP DDoS Flood Attacks on Apache2 and Nginx Web Servers with Linux Ubuntu," 2023.
- [12] D. Litespeed, S. Kasus, and I. Di, "Analisis Perbandingan Performa Web Server Apache ," vol. 06, pp. 1145–1155, 2025.
- [13] M. Bagus, T. Cahyo, H. Setiawan, I. Ratna, and I. Astutik, "Implementation and Analysis of Cloud-Native Approaches to Enhance Scalability and Performance in Dwi Jaya Understel Workshop Information System," no. 10, pp. 365–374, 2025.

6. Penulis



Erich Ricardo
Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas, Medan,
Indonesia
Penulis merupakan mahasiswa di Universitas Katolik Santo Thomas



Angela Wita Simanullang
Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas, Medan,
Indonesia
Penulis merupakan mahasiswa di Universitas Katolik Santo Thomas



Lotar Mateus Sinaga
Dosen Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas