LAPORAN PROYEK AKHIR

MEMBANGUN WEB SERVER BERBASIS LINUX (NGINX + SSL)

Disusun untuk Memenuhi Tugas Akhir Mata Kuliah

Sistem Operasi

Dosen Pengampu:

Ferdi Chahyadi, S.Kom., M.Cs

Disusun oleh:

Kelompok Wanmut

2401020001 Amira Azza Nuradiba
2401020006 Zafira Khairunnisa
2401020017 Rashika Kasih Putri
2401020025 Nurul Syafika

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK DAN TEKNOLOGI KEMARITIMAN

UNIVERSITAS MARITIM RAJA ALI HAJI

2025/2026

KATA PENGANTAR

Puji dan syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya laporan Proyek Akhir mata kuliah Sistem Operasi ini dapat diselesaikan dengan baik. Laporan ini disusun sebagai bentuk pertanggungjawaban atas pelaksanaan proyek akhir yang berjudul *"Membangun Web Server Berbasis Linux (Nginx + SSL)"*.

Penyusunan laporan ini bertujuan untuk mendokumentasikan seluruh proses pelaksanaan proyek, mulai dari tahap perancangan, implementasi sistem, hingga pengujian web server yang telah dibangun. Kami menyadari bahwa dalam penyusunan laporan ini masih terdapat keterbatasan, baik dari segi penulisan maupun isi pembahasan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan demi penyempurnaan laporan ini.

Kami mengucapkan terima kasih kepada dosen pengampu mata kuliah Sistem Operasi serta semua pihak yang telah memberikan bimbingan dan dukungan selama proses pelaksanaan proyek dan penyusunan laporan ini. Semoga laporan proyek akhir ini dapat memberikan manfaat dan menambah wawasan bagi pembaca.

Tanjungpinang, 25 Desember 2025

Kelompok Wanmut

ABSTRAK

Web server merupakan salah satu komponen penting dalam penyediaan layanan informasi berbasis jaringan. Untuk menjamin layanan web yang stabil dan aman, diperlukan konfigurasi server yang tepat serta penerapan sistem keamanan yang memadai. Pada proyek akhir mata kuliah Sistem Operasi ini telah dilakukan pembangunan web server berbasis Linux dengan menggunakan Nginx sebagai web server utama, serta penerapan virtual host, SSL (HTTPS), dan firewall.

Proses pengerjaan proyek dimulai dari instalasi sistem operasi Linux, instalasi dan konfigurasi web server, pembuatan virtual host, penerapan SSL untuk mengamankan komunikasi data, hingga konfigurasi firewall sebagai sistem perlindungan server. Setelah proses implementasi selesai, dilakukan pengujian untuk memastikan layanan web dapat diakses melalui HTTP dan HTTPS serta firewall berfungsi sesuai dengan aturan yang ditetapkan.

Hasil dari proyek akhir ini menunjukkan bahwa web server berbasis Linux dapat berjalan dengan baik dan aman sesuai dengan kebutuhan yang telah ditentukan. Melalui proyek ini, mahasiswa memperoleh pemahaman praktis mengenai administrasi server, pengelolaan layanan web, serta penerapan keamanan dasar pada sistem operasi Linux.

Kata kunci: Linux Server, Web Server,Nginx, SSL, Firewall.

DAFTAR ISI

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi saat ini menuntut ketersediaan layanan jaringan yang stabil, aman, dan mudah diakses. Salah satu layanan yang paling banyak digunakan adalah layanan web, yang berfungsi sebagai media penyampaian informasi dan komunikasi melalui jaringan internet. Agar layanan web dapat berjalan dengan baik, diperlukan sebuah web server yang dikonfigurasi secara tepat dan aman.

Sistem operasi Linux merupakan salah satu sistem operasi yang banyak digunakan sebagai server karena bersifat open source, stabil, ringan, serta memiliki tingkat keamanan yang baik. Linux juga mendukung berbagai aplikasi web server populer seperti Nginx yang mampu menangani permintaan web secara efisien. Selain itu, aspek keamanan menjadi hal yang sangat penting dalam penyediaan layanan web, terutama dalam melindungi data yang dikirimkan antara server dan pengguna.

Oleh karena itu, dalam proyek akhir mata kuliah Sistem Operasi ini telah dilakukan pembangunan web server berbasis Linux dengan menerapkan konfigurasi Nginx, virtual host, SSL (HTTPS), serta firewall. Proyek ini bertujuan untuk memberikan pemahaman praktis mengenai pengelolaan server, penerapan keamanan dasar, serta pengujian layanan web yang berjalan pada sistem operasi Linux.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah dalam proyek akhir ini sebagai berikut:

1. Bagaimana proses instalasi dan konfigurasi web server berbasis Linux?
2. Bagaimana penerapan virtual host pada web server agar dapat melayani beberapa domain?
3. Bagaimana cara mengimplementasikan SSL untuk meningkatkan keamanan akses web?
4. Bagaimana konfigurasi firewall untuk melindungi web server dari akses yang tidak diizinkan?

1.3 Tujuan Proyek

Tujuan dari pelaksanaan proyek akhir ini adalah:

1. Membangun dan mengonfigurasi web server berbasis Linux.
2. Mengimplementasikan virtual host pada web server.
3. Menerapkan SSL (HTTPS) untuk mengamankan komunikasi data.
4. Mengonfigurasi firewall sebagai sistem keamanan server.
5. Melakukan pengujian terhadap web server yang telah dibangun.

1.4 Manfaat

Manfaat yang diperoleh dari pelaksanaan proyek akhir ini antara lain:

1. Memberikan pemahaman praktis mengenai administrasi server Linux.
2. Menambah pengetahuan tentang konfigurasi web server Nginx.
3. Melatih penerapan konsep keamanan jaringan melalui SSL dan firewall.
4. Menjadi bekal keterampilan dasar dalam pengelolaan server di dunia kerja.

1.5 Batasan Masalah

Agar pembahasan dalam laporan ini lebih terarah, maka batasan masalah dalam proyek akhir ini adalah:

1. Sistem operasi yang digunakan adalah Linux.
2. Web server yang digunakan adalah Nginx.
3. SSL yang diterapkan menggunakan sertifikat self-signed.
4. Firewall yang digunakan adalah UFW atau iptables.
5. Pengujian dilakukan pada lingkungan lokal atau jaringan terbatas.

BAB II
LANDASAN TEORI

2.1 Sistem Operasi Linux

Linux merupakan sistem operasi berbasis open source yang banyak digunakan sebagai server karena memiliki stabilitas tinggi, keamanan yang baik, serta dukungan komunitas yang luas. Linux mampu berjalan pada berbagai jenis perangkat keras dan menyediakan berbagai fitur yang mendukung pengelolaan sistem server, seperti manajemen proses, manajemen memori, serta pengaturan jaringan.

Dalam lingkungan server, Linux sering digunakan untuk menyediakan layanan jaringan, termasuk layanan web. Keunggulan Linux sebagai server terletak pada kemampuannya untuk berjalan dalam waktu lama tanpa gangguan serta kemudahan dalam melakukan konfigurasi dan pemeliharaan sistem.

2.2 Web Server

Web server merupakan perangkat lunak yang berfungsi untuk menerima permintaan (request) dari klien melalui protokol HTTP atau HTTPS dan mengirimkan respons berupa halaman web atau data lainnya. Web server menjadi komponen utama dalam sistem layanan berbasis web karena berperan sebagai penghubung antara pengguna dan data yang disediakan server.

Dalam praktiknya, web server tidak hanya menampilkan halaman statis, tetapi juga mampu melayani konten dinamis serta mengelola banyak koneksi pengguna secara bersamaan. Oleh karena itu, pemilihan web server yang tepat sangat berpengaruh terhadap kinerja dan stabilitas sistem. Pada proyek akhir ini digunakan web server Nginx karena terbukti andal dan banyak digunakan pada sistem operasi Linux dan lebih efisien**,** cepat, dan stabil, terutama untuk aplikasi web modern.

2.3 Nginx

Nginx merupakan web server yang dirancang dengan arsitektur event-driven, sehingga mampu menangani banyak koneksi secara bersamaan dengan penggunaan

sumber daya yang lebih efisien. Nginx dikenal memiliki performa tinggi dan stabil, terutama pada server dengan trafik yang cukup padat.

Selain berfungsi sebagai web server, Nginx juga dapat digunakan sebagai reverse proxy dan load balancer. Kemampuan tersebut membuat Nginx sering digunakan pada sistem web modern. Dalam proyek akhir ini, Nginx menjadi salah satu alternatif web server yang dapat digunakan, tergantung pada kebutuhan dan skenario implementasi.

2.4 Virtual Host

Virtual host adalah fitur pada web server yang memungkinkan satu server menjalankan lebih dari satu website atau domain dengan konfigurasi yang terpisah. Dengan virtual host, setiap website dapat memiliki direktori penyimpanan, domain, dan pengaturan server yang berbeda meskipun berada dalam satu mesin server.

Penerapan virtual host sangat membantu dalam pengelolaan layanan web karena dapat menghemat sumber daya server dan mempermudah administrasi. Pada proyek akhir ini, virtual host digunakan untuk membuktikan bahwa satu web server Linux mampu melayani lebih dari satu layanan web secara terpisah.

2.5 SSL dan HTTPS

SSL (Secure Socket Layer) merupakan teknologi keamanan yang digunakan untuk mengenkripsi data yang dikirimkan antara klien dan server. Dengan adanya SSL, informasi seperti data login atau pertukaran data lainnya menjadi lebih aman dari risiko penyadapan.

HTTPS adalah versi aman dari protokol HTTP yang menggunakan SSL sebagai lapisan keamanannya. Penerapan HTTPS pada web server bertujuan untuk meningkatkan keamanan komunikasi data serta memberikan perlindungan tambahan bagi pengguna. Pada proyek akhir ini, SSL diterapkan untuk memastikan bahwa layanan web berjalan dengan koneksi yang lebih aman.

2.6 Firewall

Firewall adalah sistem keamanan jaringan yang berfungsi untuk mengontrol lalu lintas data yang masuk dan keluar dari server berdasarkan aturan tertentu. Firewall bertujuan untuk mencegah akses tidak sah dan melindungi server dari potensi serangan jaringan.

Pada sistem operasi Linux, firewall dapat dikonfigurasi menggunakan UFW atau iptables. Dengan penerapan firewall, hanya port dan layanan yang diperlukan saja yang diizinkan untuk diakses, sehingga risiko gangguan keamanan pada web server dapat diminimalkan. Firewall menjadi salah satu komponen penting dalam menjaga keamanan server pada proyek akhir ini.

BAB III
METODOLOGI DAN TAHAPAN PROYEK

3.1 Metode Pelaksanaan Proyek

Metode yang digunakan dalam pelaksanaan proyek akhir ini adalah metode praktikum dan implementasi langsung. Metode ini dipilih karena sesuai dengan karakteristik mata kuliah Sistem Operasi yang menekankan pemahaman konsep melalui praktik. Seluruh tahapan proyek dilakukan secara bertahap, dimulai dari persiapan sistem, instalasi server, konfigurasi layanan, hingga pengujian dan dokumentasi hasil.

3.2 Tahapan Pelaksanaan Proyek

Pelaksanaan proyek akhir ini dilakukan melalui beberapa tahapan yang disusun secara sistematis berdasarkan progres mingguan yang telah dilaksanakan.

1. **Minggu 11 – Penyusunan Proposal**

   Pada tahap ini dilakukan penyusunan proposal proyek akhir yang berisi latar belakang, tujuan, ruang lingkup, serta rencana pelaksanaan proyek. Proposal ini digunakan sebagai acuan dalam pelaksanaan proyek serta sebagai persetujuan awal terhadap topik yang dikerjakan.
2. **Minggu 12 – Instalasi Linux dan Web Server**

   Pada tahap ini dilakukan instalasi sistem operasi Linux sebagai server utama. Setelah sistem operasi berhasil terpasang, dilakukan instalasi web server Nginx beserta paket pendukung lainnya. Tahap ini bertujuan untuk menyiapkan lingkungan server yang siap digunakan untuk konfigurasi lanjutan.
3. **Minggu 13 – Konfigurasi Virtual Host dan SSL**

   Pada tahap ini dilakukan konfigurasi virtual host untuk mengelola lebih dari satu layanan web dalam satu server. Selain itu, dilakukan penerapan SSL untuk mengamankan koneksi antara klien dan server dengan menggunakan protokol HTTPS. Konfigurasi ini memastikan layanan web dapat diakses dengan aman dan terstruktur.
4. **Minggu 14 – Pengujian Server dan Firewall**

Pada tahap ini dilakukan pengujian terhadap web server dan konfigurasi firewall. Pengujian meliputi akses HTTP dan HTTPS, pengecekan virtual host, serta pengujian aturan firewall untuk memastikan hanya port yang diizinkan yang dapat diakses. Tahap ini bertujuan memastikan sistem berjalan dengan baik dan aman.

3.3 Alat dan Bahan

Alat dan bahan yang digunakan dalam pelaksanaan proyek akhir ini meliputi perangkat keras dan perangkat lunak sebagai berikut:

1. Laptop atau komputer sebagai server
2. Sistem operasi Linux
3. Web server Nginx
4. OpenSSL atau Certbot untuk SSL
5. Firewall UFW atau iptables
6. Web browser sebagai alat pengujian

3.4 Teknik Pengujian

Teknik pengujian dilakukan dengan cara mengakses layanan web melalui browser menggunakan protokol HTTP dan HTTPS. Selain itu, dilakukan pengecekan status layanan web server serta pengujian firewall untuk memastikan keamanan server. Hasil pengujian dicatat dan digunakan sebagai bahan analisis pada tahap pembahasan.

3.5 Dokumentasi Proyek

Selama pelaksanaan proyek, dilakukan dokumentasi berupa pencatatan langkah-langkah konfigurasi, pengambilan tangkapan layar (screenshot), serta penyimpanan file konfigurasi. Dokumentasi ini digunakan sebagai bukti pelaksanaan proyek dan sebagai pendukung dalam penyusunan laporan akhir.

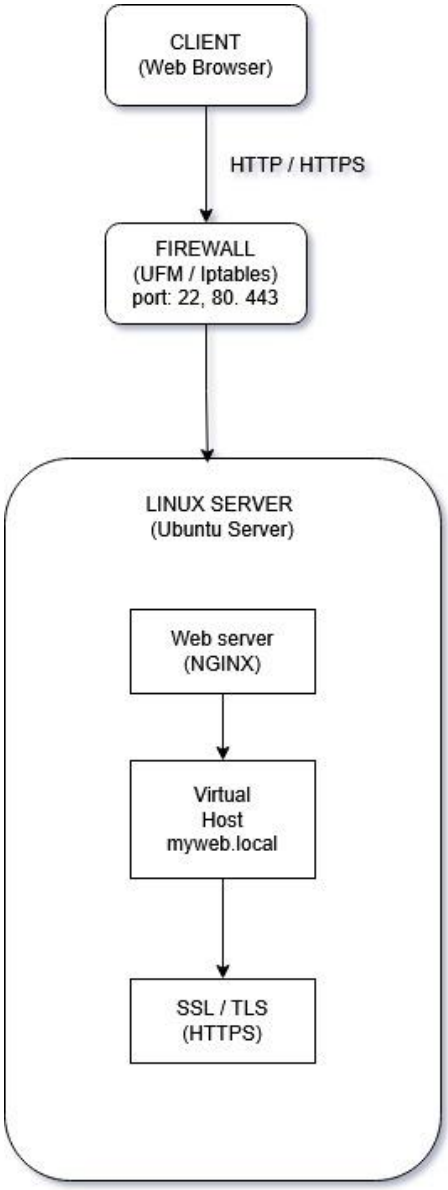
BAB IV
PERANCANGAN ARSITEKTUR

4.1 Diagram Arsitektur Sistem

Diagram arsitektur sistem menggambarkan struktur dan alur kerja web server berbasis Linux yang dibangun pada proyek akhir ini. Arsitektur sistem menggunakan pendekatan client-server, di mana klien mengakses layanan web melalui web browser menggunakan protokol HTTP atau HTTPS.

Permintaan dari klien terlebih dahulu melewati firewall yang berfungsi sebagai pengaman jaringan. Firewall dikonfigurasi untuk mengizinkan akses hanya pada port tertentu, yaitu port 22 untuk SSH, port 80 untuk HTTP, dan port 443 untuk HTTPS. Setelah melewati firewall, permintaan diteruskan ke server Linux yang berfungsi sebagai web server utama.

Server Linux menjalankan layanan Nginx sebagai web server untuk menangani permintaan web dari klien. Pada server ini diterapkan virtual host sehingga satu server dapat melayani beberapa website atau domain yang berbeda. Selain itu, layanan web juga dilengkapi dengan SSL untuk mendukung akses HTTPS sehingga komunikasi data antara klien dan server menjadi lebih aman.



Gambar 1. Diagram Arsitektur Sistem

4.2 Deskripsi Arsitektur Sistem

Arsitektur sistem pada proyek akhir ini dirancang untuk menghasilkan sistem web server yang stabil, aman, dan mudah dikelola. Penggunaan satu server Linux memungkinkan pengelolaan sistem dilakukan secara terpusat, sementara penerapan virtual host membuat server mampu melayani lebih dari satu domain secara efisien.

Dari sisi keamanan, penerapan firewall dan SSL menjadi bagian penting dalam arsitektur sistem. Firewall membatasi lalu lintas jaringan yang masuk ke server, sedangkan SSL berfungsi untuk mengenkripsi komunikasi data antara klien dan server. Kombinasi kedua mekanisme tersebut diharapkan dapat meningkatkan keamanan dan keandalan layanan web yang disediakan.

Secara keseluruhan, perancangan arsitektur sistem ini telah disesuaikan dengan kebutuhan proyek akhir dan mendukung proses implementasi serta pengujian sistem web server berbasis Linux.

4.3 Komponen Sistem

No	Komponen Sistem	Fungsi
1	Client (Web Browser)	Digunakan oleh pengguna untuk mengakses layanan web melalui protokol HTTP atau HTTPS. Client mengirimkan permintaan (request) ke server dan menerima respons berupa halaman web.
2	Jaringan	Berfungsi sebagai media penghubung antara client dan server sehingga memungkinkan terjadinya komunikasi data.
3	Firewall (UFW / Iptables)	Mengontrol lalu lintas jaringan yang masuk dan keluar dari server dengan membatasi port yang dapat diakses, sehingga meningkatkan keamanan sistem.
4	Server Linux (Ubuntu Server)	Berfungsi sebagai server utama yang menjalankan sistem operasi Linux untuk mengelola layanan web, konfigurasi jaringan, dan keamanan server.
5	Web Server (Nginx)	Bertugas menerima permintaan dari client dan mengirimkan respons berupa halaman web, serta mengelola layanan web pada server.
6	Virtual Host	Digunakan untuk memisahkan layanan website berdasarkan domain atau direktori tertentu sehingga satu server dapat melayani beberapa website secara bersamaan.

7	SSL	Berfungsi untuk mengamankan komunikasi data antara client dan server dengan cara mengenkripsi data, sehingga akses web dapat dilakukan melalui protokol HTTPS.

BAB V
IMPLEMENTASI DAN KONFIGURASI

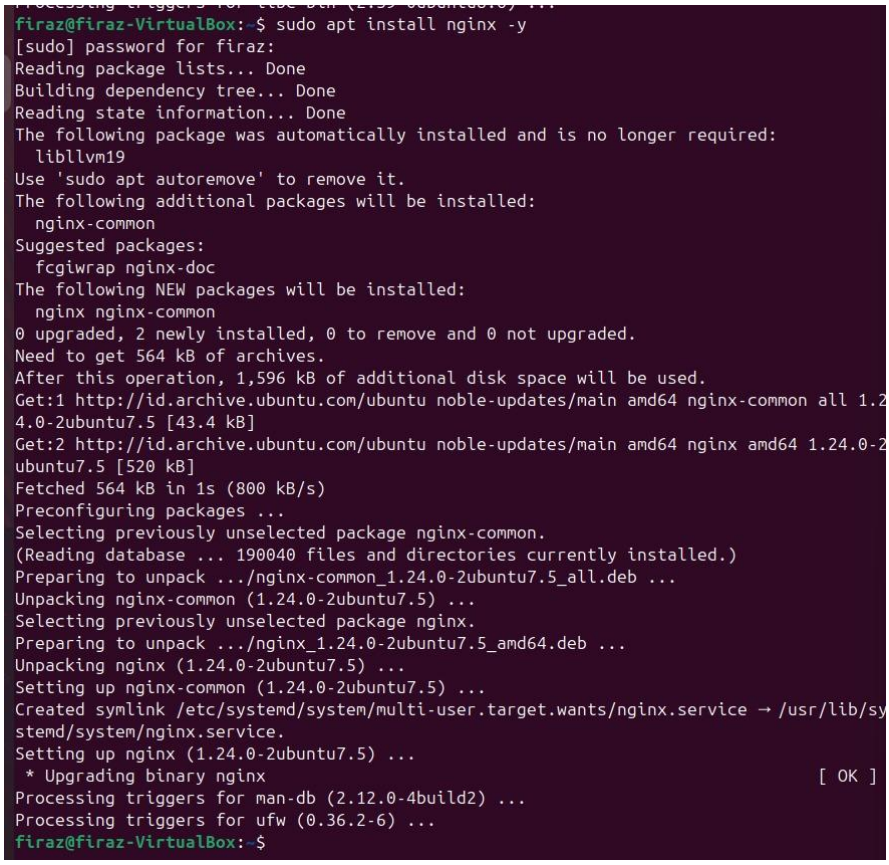
5.1 Instalasi Linux Server, Nginx, domain lokal, SSL self-signed

Pada tahap ini dilakukan instalasi sistem operasi Linux Server sebagai platform utama server. Setelah sistem operasi berhasil diinstal, dilakukan pemasangan web server Nginx untuk menyediakan layanan web. Selanjutnya, dilakukan pengaturan domain lokal untuk keperluan pengujian akses website pada lingkungan lokal. Selain itu, diterapkan SSL self-signed untuk mengaktifkan akses HTTPS sebagai langkah awal dalam penerapan keamanan komunikasi data pada web server.

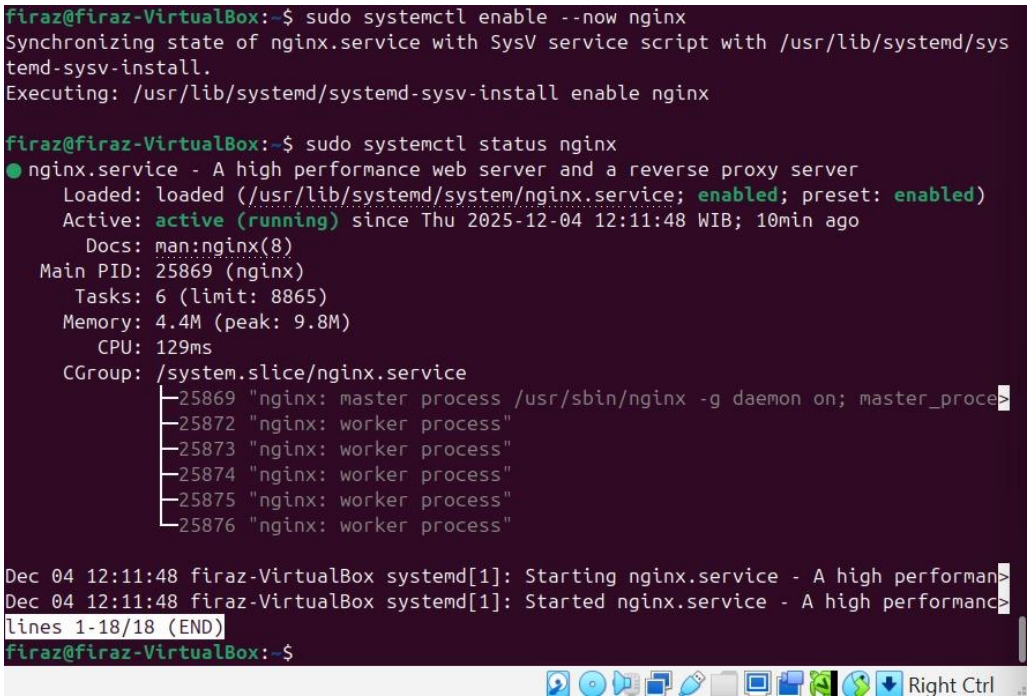
1. **Instalasi Linux Server**

```
firaz@firaz-VirtualBox:~$ sudo apt update && sudo apt upgrade -y
[sudo] password for firaz:
Hit:1 http://id.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://id.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://id.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
171 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  libllvm20 linux-headers-6.14.0-36-generic linux-hwe-6.14-headers-6.14.0-36
  linux-hwe-6.14-tools-6.14.0-36 linux-image-6.14.0-36-generic
  linux-modules-6.14.0-36-generic linux-modules-extra-6.14.0-36-generic
  linux-tools-6.14.0-36-generic
The following packages will be upgraded:
  bind9-dnsutils bind9-host bind9-libs bluez bluez-cups bluez-obexd cloud-init
  coreutils cups cups-bsd cups-client cups-common cups-core-drivers cups-daemon
  cups-filters cups-filters-core-drivers cups-ipp-utils cups-ppdc cups-server-common
  dconf-cli dconf-gsettings-backend dconf-service distro-info-data dpkg
  evolution-data-server evolution-data-server-common firmware-sof-signed fwupd gdm3
  ghostscript gir1.2-gdm-1.0 gir1.2-gtk-4.0 gir1.2-javascriptcoregtk-4.1
  gir1.2-javascriptcoregtk-6.0 gir1.2-nm-1.0 gir1.2-webkit-6.0 gir1.2-webkit2-4.1
  gnome-shell-extension-desktop-icons-ng intel-microcode libbluetooth3 libc-bin
  libc-dev-bin libc-devtools libc6 libc6-dbg libc6-dev libcamel-1.2-64t64 libcups2t64
  libcupsfilters2-common libcupsfilters2t64 libcupsimage2t64 libdconf1
  libebackend-1.2-11t64 libebook-1.2-21t64 libebook-contacts-1.2-4t64 libecal-2.0-3
  libedata-book-1.2-27t64 libedata-cal-2.0-2t64 libedataserver-1.2-27t64
  libedataserverui-1.2-4t64 libegl-mesa0 libfwupd2 libgbm1 libgdm1 libgl1-mesa-dri
  libglx-mesa0 libgs-common libgs10 libgs10-common libgtk-4-1 libgtk-4-bin
  libgtk-4-common libgtk-4-media-gstreamer libgtop-2.0-11 libgtop2-common
  libipa-hbac0t64 libjavascriptcoregtk-4.1-0 libjavascriptcoregtk-6.0-1 libldb2
  libmalcontent-0-0 libnm0 libnss-sss libnss-systemd libopenjp2-7 libpam-modules
  libpam-modules-bin libpam-runtime libpam-sss libpam-systemd libpam0g
  libpoppler-cpp0t64 libpoppler-glib8t64 libpoppler134 libpython3-stdlib
  libpython3.12-minimal libpython3.12-stdlib libpython3.12t64 libsmbclient0 libssh-4
  libssl3t64 libsss-certmap0 libsss-idmap0 libsss-nss-idmap0 libsystemd-shared
```

2. Instalasi Nginx



3. Mengaktifkan dan mengecek status aktif Nginx



4. Mengelolah atau menambahkan domain lokal

```
firaz@firaz-VirtualBox:~$ sudo nano /etc/hosts
[sudo] password for firaz:
```

Yang dimana setelah berhasil menambahkan domain lokal dan masuk ke domain kita lanjut ke tahapan:

a) menambahkan ip virtual box dan juga ip localhostb)

```
127.0.0.1 localhost
127.0.1.1 firaz-VirtualBox
127.0.0.1 myweb.local
192.168.56.1 myweb.local

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

b) lakukan konfigurasi pada Nginx

```
firaz@firaz-VirtualBox:~$ sudo nano /etc/nginx/sites-available/myweb.local
firaz@firaz-VirtualBox:~$ sudo ln -s /etc/nginx/sites-available/myweb.local /etc/nginx/sites-enabled/
ln: failed to create symbolic link '/etc/nginx/sites-enabled/myweb.local': File exists
firaz@firaz-VirtualBox:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
firaz@firaz-VirtualBox:~$ sudo systemctl reload nginx
```

5. SSL self-signed

```
firaz@firaz-VirtualBox:~$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/myweb.key -out /etc/ssl/certs/myweb.crt
.+.....+.....+.....+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.+.......+.......+...+..+.+..
.......+....+.....+..+...+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.+..+...+.......+......+..+...
+...+.+.....+......+....+...+..+......+...+....+.....+......+.........+.....+.........+..............+......+......+..........+.
....+..........+......+.....+.............+......+...+..................+..+.+..........+.+...+.....+.........+.....+....+.....
..+...+.+...+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
.+......+.........+.......+...+..+...+.............+...+.+.....+.+......+..+......+.............+...+....+...+...+..+.+.......+..+...+
...+...+.+......++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.+...+.......+.........+....+...+.........+......
...+......+...+...++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*...+......+..+...................+...........+
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ID
State or Province Name (full name) [Some-State]:Kepulauan Riau
Locality Name (eg, city) []:Tanjung Pinang
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UMRAH
Organizational Unit Name (eg, section) []:Kelompok Wanmut
Common Name (e.g. server FQDN or YOUR name) []:myweb.local
Email Address []:wanmut@gmail.com
```

5.2 Konfigurasi Virtual Host dan HTTPS serta Penataan Struktur Web

Tahap ini difokuskan pada konfigurasi virtual host agar web server mampu melayani lebih dari satu website atau domain pada satu server. Setiap virtual host diatur dengan direktori dan konfigurasi terpisah sehingga struktur website menjadi lebih rapi dan terorganisir. Selain itu, dilakukan pengaktifan HTTPS pada virtual host menggunakan sertifikat SSL yang telah dibuat sebelumnya, sehingga website dapat diakses melalui koneksi yang lebih aman.

1. Membuat dan mengkonfigurasikan virtual host

```
firaz@firaz-VirtualBox:~$ sudo nano /etc/nginx/sites-available/myweb.local
  GNU nano 7.2                                  /etc/nginx/sites-available/myweb.local
server {
    listen 80;
    server_name myweb.local;

    root /var/www/myweb.local;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}

server {
    listen 443 ssl;
    server_name myweb.local;

    ssl_certificate /etc/ssl/certs/myweb.crt;
    ssl_certificate_key /etc/ssl/private/myweb.key;

    root /var/wwww/myweb.local;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

a) Tes konfigurasi virtual host

```
firaz@firaz-VirtualBox:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
firaz@firaz-VirtualBox:~$ sudo systemctl reload nginx
```

2. Konfigurasi HTTPS (SSL Self-Signed)

Pada tahap ini, konfigurasi HTTPS dilakukan dengan menggunakan sertifikat SSL self-signed yang telah dibuat pada tahap sebelumnya. Sertifikat tersebut digunakan

kembali pada konfigurasi virtual host Nginx untuk mengaktifkan akses HTTPS melalui port 443 pada domain lokal myweb.local. Terdapat pada isi file virtual host, yaitu bagian:

```
    ssl_certificate /etc/ssl/certs/myweb.crt;
    ssl_certificate_key /etc/ssl/private/myweb.key;
```

3. Penataan struktur web

```
firaz@firaz-VirtualBox:/var/www/myweb.local$ ls -l /var/www/myweb.local
total 8
drwxrwxr-x 2 www-data www-data 4096 Dec 13 06:00 css
-rwxrwxr-x 1 www-data www-data  722 Dec 13 05:56 index.html
firaz@firaz-VirtualBox:/var/www/myweb.local$ ls -l css
total 4
-rwxrwxr-x 1 www-data www-data 292 Dec 13 06:00 style.css
firaz@firaz-VirtualBox:/var/www/myweb.local$
```

Screenshot di atas menunjukkan struktur direktori website pada /var/www/myweb.local yang terdiri dari file index.html sebagai halaman utama serta direktori css yang berisi file style.css. Penataan struktur ini dilakukan untuk memisahkan antara konten HTML dan pengaturan tampilan (CSS) agar website lebih terorganisir dan mudah dikelola.

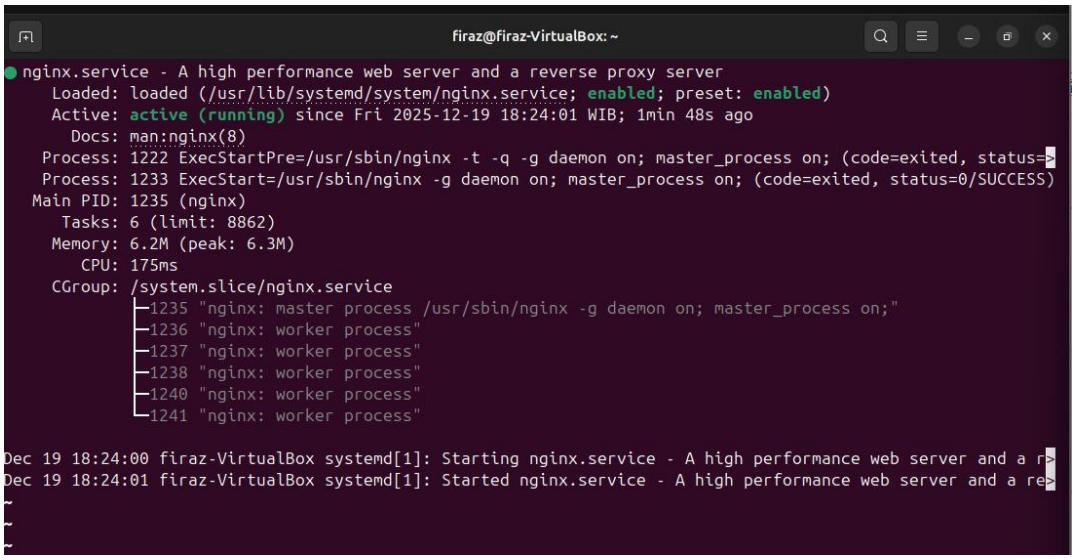
BAB VI
PENGUJIAN DAN ANALISIS

6.1 Pengujian Sistem (Akses Publik, Firewall, dan Load Test Sederhana)

Pada tahap pengujian, dilakukan pengecekan akses website melalui jaringan untuk memastikan layanan web dapat diakses dengan baik. Selain itu, pengujian firewall dilakukan untuk memastikan hanya port yang diizinkan saja yang dapat diakses. Pengujian juga mencakup load test sederhana untuk melihat respon web server ketika menerima beberapa permintaan secara bersamaan. Hasil pengujian menunjukkan bahwa sistem web server berjalan dengan stabil dan sesuai dengan konfigurasi yang telah diterapkan.

1. Pengujian status web server

a) Cek nginx berjalan atau tidak, command: **sudo systemctl status nginx**



Terlihat bahwa status active (running) yang artinya nginx berjalan

b) Untuk pengujian akses http dan https sudah dilakukan di bagian implementasi

2. **Pengujian firewell (UFW)**

```
firaz@firaz-VirtualBox:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
Nginx Full                 ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
Nginx Full (v6)            ALLOW       Anywhere (v6)
```

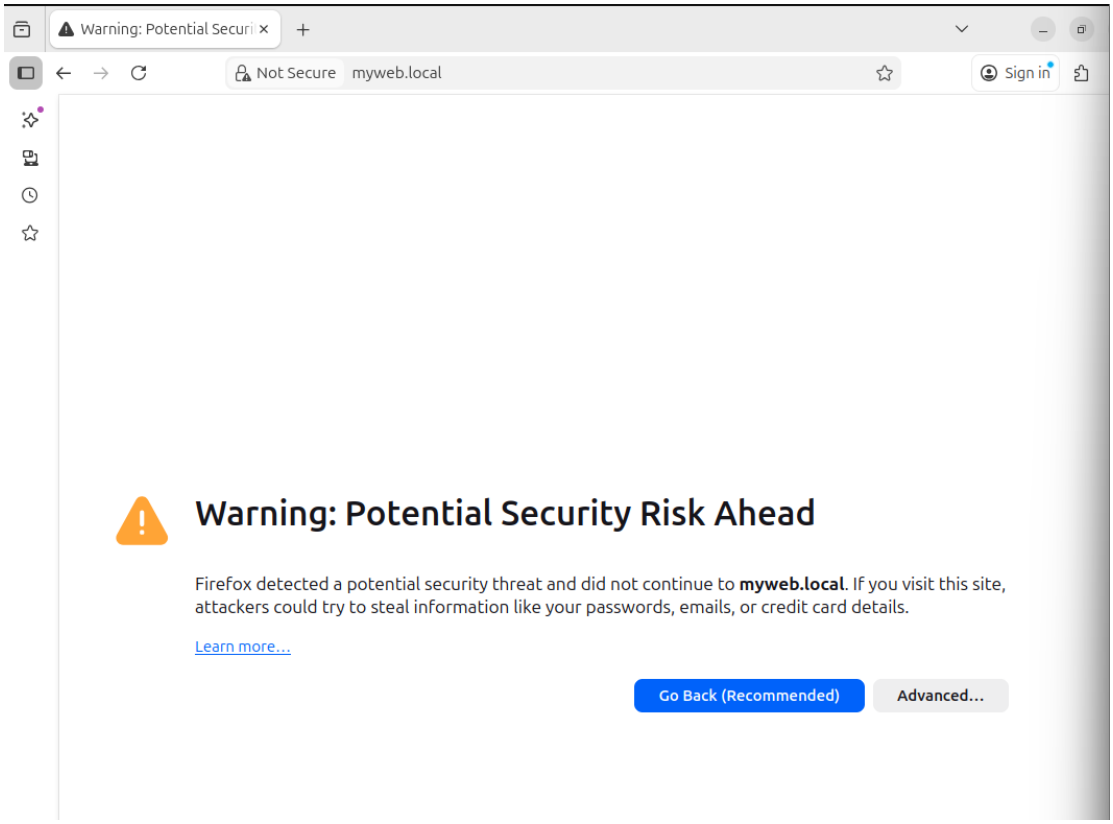
Pengujian firewall dilakukan untuk memastikan sistem keamanan berjalan dengan baik. Firewall berada dalam kondisi aktif dan hanya membuka port yang diperlukan untuk layanan web dan akses administrasi. Pada gambar terlihat bahwa status ufw sudah **active**

3. **Pengujian Virtual Host Menggunakan Protokol HTTPS (SSL Self-Signed).**

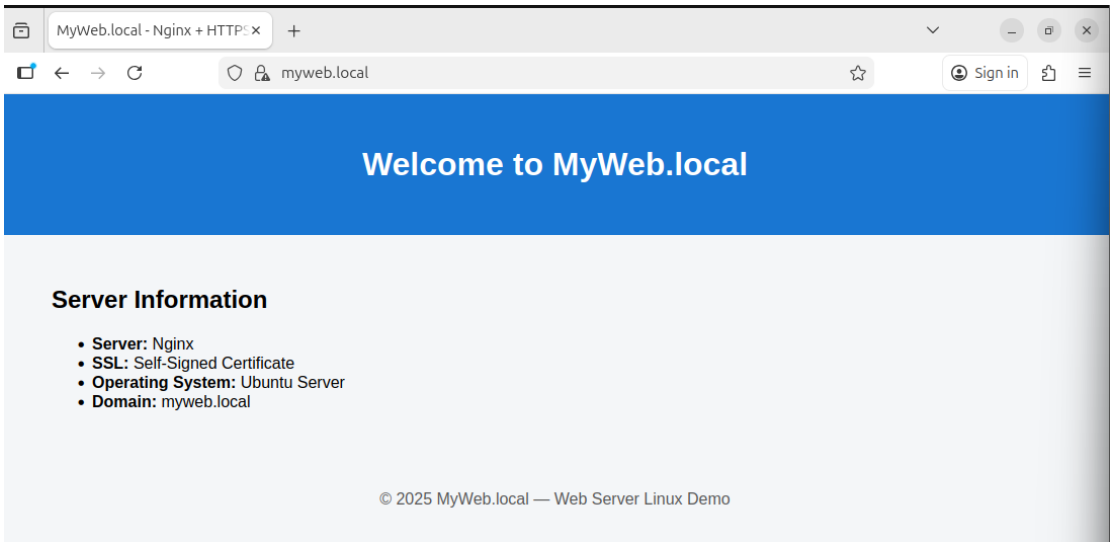
```
firaz@firaz-VirtualBox:~$ curl -k -H "Host: myweb.local" https://myweb.local

    Caution: You are using the Snap version of curl.
    Due to Snap's sandbox nature, this version has some limitations.
    For example, it may not be able to access hidden folders in your home directory
    or other restricted areas of the os.

    Which means you may encounter errors when using snap curl to download and execute some script.
    For those cases, you might want to use the native curl package.
    For details, see: https://github.com/boukendesho/curl-snap/issues/1

    To stop seeing this message, run the following command:
    $ curl.snap-acked

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>MyWeb.local - Nginx + HTTPS</title>
    <link rel="stylesheet" href="/css/style.css">
</head>
<body>

    <header class="hero">
        <h1>Welcome to MyWeb.local</h1>
    </header>

    <section class="content">
        <h2>Server Information</h2>
        <ul>
            <li><strong>Server:</strong> Nginx</li>
            <li><strong>SSL:</strong> Self-Signed Certificate</li>
            <li><strong>Operating System:</strong> Ubuntu Server</li>
            <li><strong>Domain:</strong> myweb.local</li>
        </ul>
    </section>

    <footer>
        <p>© 2025 MyWeb.local — Web Server Linux Demo</p>
    </footer>

</body>
</html>
```

Pengujian virtual host menggunakan protokol HTTPS dilakukan untuk memastikan penerapan SSL pada web server Nginx. Website dapat diakses melalui domain myweb.local menggunakan koneksi aman.

Pada akses pertama menggunakan HTTPS, browser menampilkan peringatan keamanan karena penggunaan sertifikat SSL self-signed. Setelah sertifikat diterima secara manual, akses HTTPS selanjutnya dapat dilakukan tanpa menampilkan peringatan kembali.



Server web berjalan di https

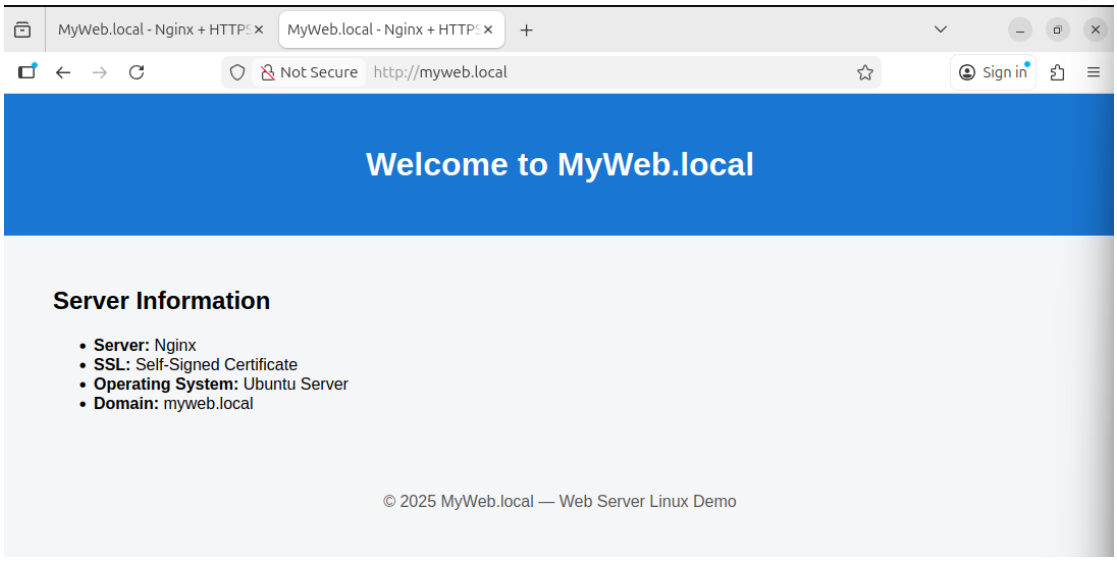


4. Pengujian Virtual Host Menggunakan Protokol HTTP.

```
firaz@firaz-VirtualBox:~$ curl -H "Host: myweb.local" http://myweb.local

    Caution: You are using the Snap version of curl.
    Due to Snap's sandbox nature, this version has some limitations.
    For example, it may not be able to access hidden folders in your home directory
    or other restricted areas of the os.

    Which means you may encounter errors when using snap curl to download and execute some script.
    For those cases, you might want to use the native curl package.
    For details, see: https://github.com/boukendesho/curl-snap/issues/1

    To stop seeing this message, run the following command:
    $ curl.snap-acked

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>MyWeb.local - Nginx + HTTPS</title>
    <link rel="stylesheet" href="/css/style.css">
</head>
<body>

    <header class="hero">
        <h1>Welcome to MyWeb.local</h1>
    </header>

    <section class="content">
        <h2>Server Information</h2>
        <ul>
            <li><strong>Server:</strong> Nginx</li>
            <li><strong>SSL:</strong> Self-Signed Certificate</li>
            <li><strong>Operating System:</strong> Ubuntu Server</li>
            <li><strong>Domain:</strong> myweb.local</li>
        </ul>
    </section>

    <footer>
        <p>© 2025 MyWeb.local — Web Server Linux Demo</p>
    </footer>

</body>
</html>
```

Pengujian virtual host dilakukan dengan mengakses domain lokal myweb.local melalui protokol HTTP menggunakan browser. Hasil pengujian menunjukkan bahwa domain berhasil mengarah ke direktori website yang telah dikonfigurasikan pada web server Nginx dan menampilkan halaman web dengan baik.

Server web berjalan di http



5. **Pengujian load test kecil**

```
firaz@firaz-VirtualBox:~$ ab -n 100 -c 10 https://myweb.local/
This is ApacheBench, Version 2.3 <$Revision: 1903618 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking myweb.local (be patient).....done


Server Software:        nginx/1.24.0
Server Hostname:        myweb.local
Server Port:            443
SSL/TLS Protocol:       TLSv1.3,TLS_AES_256_GCM_SHA384,2048,256
Server Temp Key:        X25519 253 bits
TLS Server Name:        myweb.local

Document Path:          /
Document Length:        722 bytes

Concurrency Level:      10
Time taken for tests:   0.234 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      96400 bytes
HTML transferred:       72200 bytes
Requests per second:    426.90 [#/sec] (mean)
Time per request:       23.424 [ms] (mean)
Time per request:       2.342 [ms] (mean, across all concurrent requests)
Transfer rate:          401.89 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        3   13   3.9     13      29
Processing:     1    8   4.1      8      23
Waiting:        0    6   4.6      5      23
Total:          5   22   5.9     22      39

Percentage of the requests served within a certain time (ms)
  50%     22
  66%     23
  75%     24
  80%     25
  90%     28
  95%     34
  98%     36
  99%     39
 100%     39 (longest request)
```

Pengujian load test kecil dilakukan menggunakan Apache Benchmark untuk mengetahui kemampuan server dalam menangani beberapa permintaan secara bersamaan. Pengujian dilakukan dengan total 100 request dan 10 koneksi secara bersamaan. Hasil pengujian menunjukkan bahwa seluruh request dapat diproses tanpa error, sehingga server dinyatakan stabil untuk penggunaan dasar.

Kesimpulan: Hasil pengujian menunjukkan bahwa seluruh request berhasil diproses tanpa kegagalan (failed requests = 0), sehingga server dinyatakan stabil untuk penggunaan dasar.

6. Pengujian Log

Pengujian log dilakukan sebagai pengujian tambahan untuk memastikan sistem berjalan dengan baik.

```
firaz@firaz-VirtualBox:~$ sudo tail /var/log/nginx/access.log
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:13:18 +0700] "GET / HTTP/1.0" 200 722 "-" "ApacheBench/2.3"
127.0.0.1 - - [19/Dec/2025:19:31:00 +0700] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:145.0) Gecko/20100101
 Firefox/145.0"
127.0.0.1 - - [19/Dec/2025:19:31:15 +0700] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:145.0) Gecko/20100101
 Firefox/145.0"
127.0.0.1 - - [19/Dec/2025:19:31:15 +0700] "GET /css/style.css HTTP/1.1" 304 0 "http://myweb.local/" "Mozilla/5.0 (X11; Ubuntu; Linux x
86_64; rv:145.0) Gecko/20100101 Firefox/145.0"
```

Pengujian log akses dilakukan untuk memastikan bahwa setiap aktivitas permintaan ke server tercatat dengan baik. Berdasarkan hasil pengujian, permintaan dari pengujian load test menggunakan Nginx Benchmark serta akses melalui browser berhasil tercatat pada file access log Nginx. Hal ini menunjukkan bahwa mekanisme pencatatan log berjalan dengan normal.

BAB VII
PENUTUP

7.1 Kesimpulan

Berdasarkan pelaksanaan proyek akhir yang telah dilakukan, dapat disimpulkan bahwa sistem web server berbasis Linux berhasil dibangun dan dikonfigurasi dengan baik. Instalasi sistem operasi Linux sebagai server utama, pemasangan web server Nginx, serta pengaturan domain lokal dapat berjalan sesuai dengan perencanaan yang telah ditetapkan.

Konfigurasi virtual host memungkinkan satu server melayani lebih dari satu website dengan struktur yang terorganisir. Penerapan SSL self-signed berhasil mengaktifkan layanan HTTPS sehingga komunikasi data antara klien dan server menjadi lebih aman. Selain itu, konfigurasi firewall mampu membatasi akses jaringan hanya pada port yang diperlukan, sehingga meningkatkan keamanan sistem.

Hasil pengujian menunjukkan bahwa layanan web dapat diakses dengan baik, konfigurasi firewall berjalan sesuai aturan, dan web server mampu menangani beban akses dalam pengujian sederhana. Dengan demikian, proyek akhir ini telah mencapai tujuan yang diharapkan serta memberikan pemahaman praktis mengenai pengelolaan web server dan keamanan sistem berbasis Linux.

7.2 Saran

Berdasarkan hasil pelaksanaan proyek akhir ini, sistem web server yang telah dibangun masih dapat dikembangkan lebih lanjut. Pada tahap selanjutnya, sistem dapat ditingkatkan dengan menggunakan sertifikat SSL resmi agar layanan web dapat diakses tanpa peringatan keamanan pada browser. Selain itu, pengujian beban (load testing) dapat dilakukan secara lebih mendalam untuk mengetahui kemampuan server dalam menangani jumlah pengguna yang lebih besar. Pengembangan sistem keamanan juga dapat ditingkatkan dengan menambahkan mekanisme monitoring dan logging untuk memantau aktivitas server secara real-time.

DAFTAR PUSTAKA

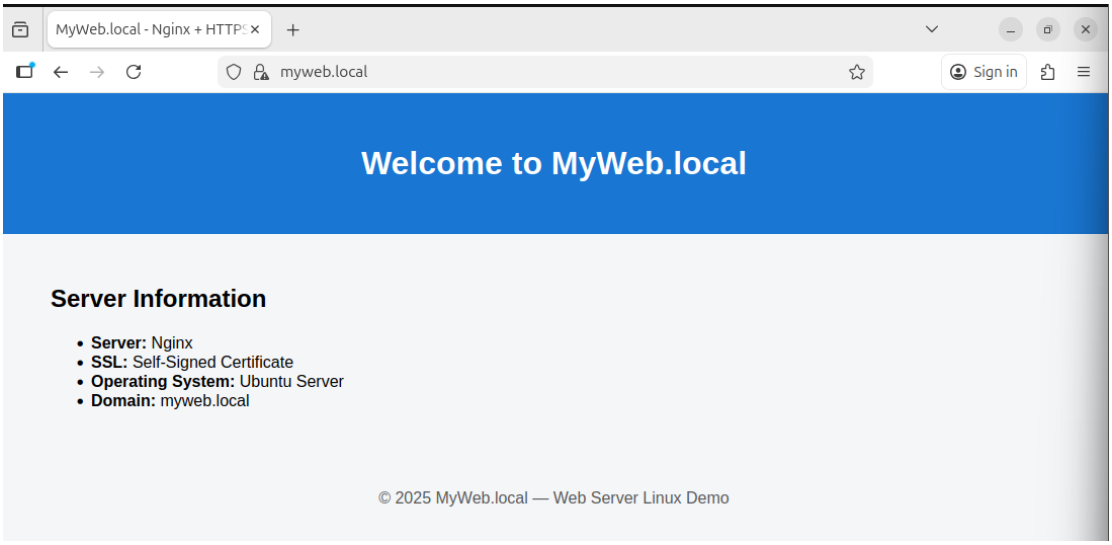
Ubuntu Documentation. (2024). *Ubuntu Server Guide*. Retrieved from https://ubuntu.com/server/docs

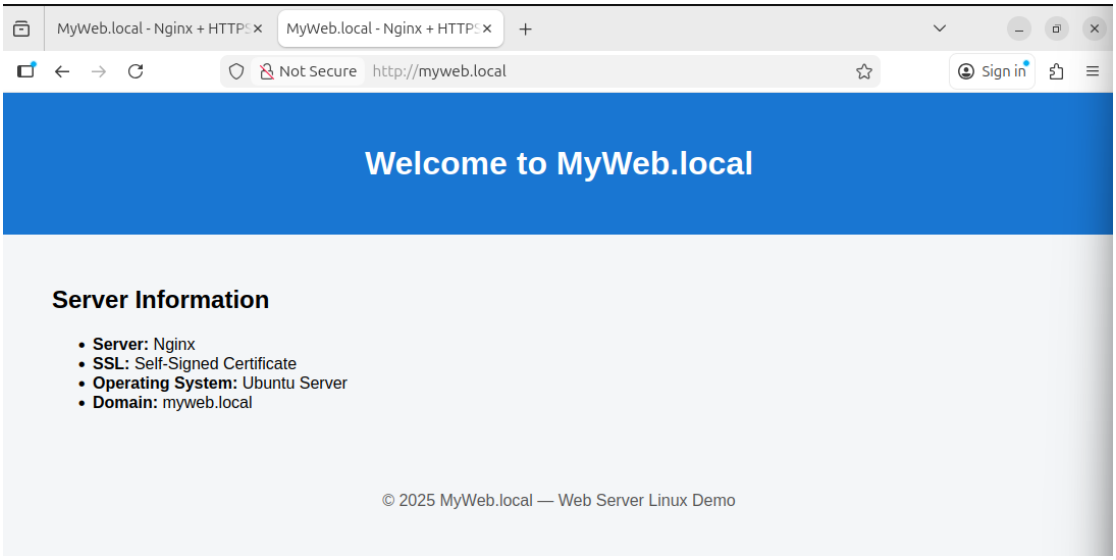
Nginx Documentation. (2024). *NGINX Official Documentation*. Retrieved from https://nginx.org/en/docs

Linux Documentation Project. (2023). *Linux System Administration Guide*. Retrieved from https://www.tldp.org

Stallings, W. (2018). *Operating Systems: Internals and Design Principles*. Retrieved from https://www.pearson.com

LAMPIRAN


MyWeb.local - Nginx + HTTPS
myweb.local
Sign in
Welcome to MyWeb.local
Server Information
Server: Nginx
SSL: Self-Signed Certificate
Operating System: Ubuntu Server
Domain: myweb.local
© 2025 MyWeb.local — Web Server Linux Demo


MyWeb.local - Nginx + HTTPS
MyWeb.local - Nginx + HTTPS
Not Secure
http://myweb.local
Sign in
Welcome to MyWeb.local
Server Information
Server: Nginx
SSL: Self-Signed Certificate
Operating System: Ubuntu Server
Domain: myweb.local
© 2025 MyWeb.local — Web Server Linux Demo