IFB-452 KOMPUTASI AWAN AA

**INTEGRASI BACKEND DENGAN FRONTEND MENGGUNAKAN NGINX**



Disusun Oleh:

Hanifah Dwi Aprilianti (152022050)

**INSTITUT TEKNOLOGI NASIONAL BANDUNG**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INFORMATIKA**

**2025**

**Integrasi Backend dengan Frontend menggunakan Nginx**

NGINX adalah web server yang juga berfungsi sebagai email proxy, reverse proxy, dan load balancer. Struktur software ini bersifat asinkron dan event-driven; yang memungkinkan banyak request atau permintaan diproses pada waktu bersamaan. Selain itu, NGINX juga bisa diskalakan.

Buatlah terlebih dahulu instance di ec2 pada AWS untuk instance frontend dan backend nya, Langkah pembuatannya telah ada di laporan sebelumnya. Lalu ikuti Langkah berikut :

|  |
| --- |
| 1. Transfer file “backend.pem” dari komputer lokal (frontend) ke server EC2 (backend). File kunci ini diperlukan untuk autentikasi dalam mengelola server backend. |
| 1. Masuk ke instance Frontend dengan syntax : *ssh -I “modul5.pem”* [*ec2-user@13.229.75.119*](mailto:ec2-user@13.229.75.119) *.* lalu ketikan syntax berikut : |
| 1. Masukkan perintah berikut untuk memasuki folder nginx nya : |
| 1. Jalankan perintah berikut pada folder nginx : |
| 1. Membuat file arsip untuk seluruh isi folder nginx\_complete menjadi satu file bernama nginx\_complete.tar.gz dalam format kompresi .tar.gz.     Cek apakah file sudah masuk |
| 1. Transfer file arsip nginx\_complete.tar.gz dari server asal ke server tujuan IP 10.0.2.87. File backend.pem digunakan sebagai kunci autentikasi untuk mengamankan proses transfer. |
| 1. Pindah ke instance backend dan masuk ke folder nginx lalu ketikan perintah berikut:   *sudo dnf install -y --disablerepo="\*" \*.rpm* |
| 1. Kemudian buatlah projek backend sederhananya, disini saya menggunakan Node.Js |
| 1. Kembali dulu ke instance frontend untuk membuat folder, lalu ketikan perintah ini : |
| 1. Pada folder yg sudah dibuat tadi install Node.Js nya, ketikan perintah berikut :   *sudo yum install -y nodejs* |
| 1. Masukkan perintah ini:     Jika sudah masukkan perintah ***npm install express*** |
| 1. Ketikan perintah *nano server.js* : lalu isi seperti ini |
| 1. Masukkan perintah *nano data.json* : lalu ketikan isi sepeerti ini |
| 1. *tar -czvf tugas.tar.gz tugasVPChani/* ketikan perintah berikut untuk membuat file tugas yg akan di zip. Maka tampil seperti ini : |
| 1. kirim file zip tadi ke backend |
| 1. masukkan syntax ini : wget <https://nodejs.org/dist/v18.19.0/node-v18.19.0-linux-x64.tar.xz> |
| 1. kirim Kembali file tadi ke backend : scp -i "backend.pem" node-v18.19.0-linux-x64.tar.xz ec2-user@10.0.2.87:~ |
| 1. masukan syntax berikut : |
| 1. start Nginx di backend pada folder tugasVPChani |
| 1. masukkan perintah : sudo nano /etc/nginx/nginx.conf   ubah code server menjadi seperti ini :  server {  listen 80;    server\_name \_;      # Load configuration files for the default server block.  include /etc/nginx/default.d/\*.conf;    location /api/ {  proxy\_pass http://localhost:3000/;  proxy\_http\_version 1.1;  proxy\_set\_header Upgrade $http\_upgrade;  proxy\_set\_header Connection 'upgrade';  proxy\_set\_header Host $host;  proxy\_cache\_bypass $http\_upgrade;  }  error\_page 404 /404.html;  location = /404.html {  }    error\_page 500 502 503 504 /50x.html;  location = /50x.html {  }  } |
| 1. sudo yum install httpd 2. cd /var/www/html |
| 1. lalu ketik perintah : sudo nano index.html |
| 1. cobalah buka ip frontend pada browser     Jika tampil seperti ini maka frontend berhasil mengambil dan menampilkan data dari backend melalui koneksi IP public frontend. Data yang diterima berupa format JSON, yang kemudian ditampilkan di halaman browser. Hal ini menandakan bahwa integrasi antara frontend dan backend menggunakan Nginx berhasil. |