

VPS Server Setup Guide: PostgreSQL & Node.js on Subdomain

1. Server Access & Preparation

- SSH into your VPS: `ssh root@your-server-ip`
- Update packages: `sudo apt update && sudo apt upgrade -y`

2. PostgreSQL Setup

- Install PostgreSQL: `sudo apt install postgresql postgresql-contrib -y`
- Switch to postgres user: `sudo -i -u postgres`
- Create database: `createdb yourdbname`
- Create user: `createuser --interactive yourdbuser`
- Set password: `psql -c "ALTER USER yourdbuser WITH PASSWORD 'yourpassword';"`
- Edit `pg_hba.conf`: `nano /etc/postgresql/14/main/pg_hba.conf`
- Restart service: `sudo systemctl restart postgresql`

3. Node.js Setup

- Install Node.js setup: `curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -`
- Install Node.js & npm: `sudo apt install -y nodejs`
- Check versions: `node -v && npm -v`
- Install PM2: `sudo npm install -g pm2`
- Start app: `pm2 start app.js --name myapp`
- Enable startup: `pm2 startup systemd && pm2 save`

4. Apache/Nginx Reverse Proxy for Subdomain

- Apache config example: `/etc/apache2/sites-available/app.example.com.conf`
- ServerName `app.example.com` ProxyPass `/ http://localhost:3000/` ProxyPassReverse `/ http://localhost:3000/`
- Enable modules & site: `sudo a2enmod proxy proxy_http && sudo a2ensite app.example.com.conf`
- Restart Apache: `sudo systemctl restart apache2`
- Enable SSL (optional): `sudo certbot --apache -d app.example.com`

5. Testing

- Check PostgreSQL: `psql -U yourdbuser -d yourdbname -h localhost -W`
- Open in browser: `http://app.example.com`
- Verify Node.js app is loading correctly