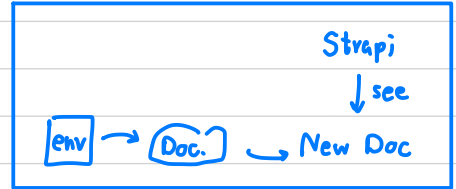
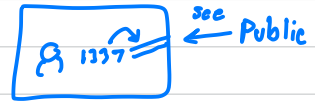


สรุป

- ได้เว็บมา 1 เร็น wd22.cloud-workshop.online/api/books , [/admin/](http://wd22.cloud-workshop.online/admin/) (login Admin)
- เขียนเว็บใช้ axios (เขียน API) , Deploy หน้า React+Strapi ใน 1 url
- strapi-react-deploy (git)
- introduction → Deploy (ใช้ env หรือ)
- ใช้ Docker ใน strapi ผ่าน environment ใน Docker
- server/config/database.js → ดึง env ใน Docker
- Docker ผ่าน env ได้ทันที (วิธีอื่น 10)
- หน้า Server ใน cloud , Docker has private NET.
- map port มินิมา 1337 → ใช้ strapi ใน Docker (Container)
127.0.0.1 → ip ใน Host (เชื่อมกับ Host)

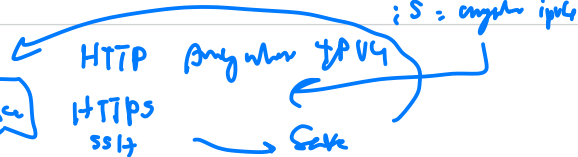


Host : 1337



- AWS(aws) → EC2 → EC2 Dashboard (Resource)
- หน้า instances (Region ใหม่ , Singapore) * ~~✖~~
- Network & Sec → Elastic IPs (ขอ ip ใหม่) → Allocate ip address → IPV4
ให้ใช้ชื่อ id Strapi Web (Gen key ใหม่)
- Instances → Launch Instance → Name (Strapi Server) → Quick Start (ubuntu) → 22.04 LTS
Instance type → t2.micro (free) ใหม่ → t2.small (รัน OK) → Key Pair → create new
(Choco git → Linux Command)
- ssh-keygen -t rsa -b 2048 (RSA No Protect)
ssh-keygen -t rsa -b 2048 → Terminal → Yes → .ssh → id-ed25519.pub } → ให้ใช้ชื่อ
User → ubuntu → .ssh id-ed25519 } *
--perm's "/.ssh/AndOnlyToolsOnPath"
- .pub → key pair → ed
- Network → input key pair → paste (.pub) (Do First)
- Firewall → หน้า Secur Group → create (Web Server) (Name) : Inbound rules : port 80
* : S = open ip

Storage : 1x20 GB → Launch Instance



Instance

Name

vbuntu

E2.small

key pair

Edit Network → Auto assign → Disable

Secure Group

20 GB

Pin.

Action → Manage IP Address

Associate Instance

(lastic ED-Instance → v1.1 IPV4 via Instance network settings

• ssh vbuntu @ IPaddress

• sudo su - , exit → to be root

vbuntu sudo cp .ssh/authorized_keys /root/.ssh/

Install Docker (as User)

• Copy → Paste → ①

• ②

x Certbot Install → py3

- Clone - react - repo via info

- cd prod/ (bg run)

- docker compose up -d

- cp ../server/.env .

- pico .env

API-Token

SALT

SECRET

SALT

SECRET

} re , change pass (optional)

- Copy read me → .env & ~ (for Sqlite & on)
- docker compose up -d
- Server → build in cloud, Cloud → built in can → copy paste

2V Server

Client

- cd client

- npm install

App.js: http://localhost → { config.serverUrl Prefix

config.js → let's wd & server by **run in Web**

- npm run build → get dir Build

- scp -r build root@ip server [IPV4] : var/www/client

• prod # cd /var/www (linux) cp ./* , scp : into server

• mk dir /var/www } change → server

prod # apt install nginx

Client cd prod # nginx.conf → let's wd 22 14 server_name

cp nginx.conf /etc/nginx/sites-available/app.conf

ln -s /etc/nginx/sites-available/app.conf /etc/nginx/sites-enabled/

↓
create symbolic link

nginx -t , restart

systemctl reload nginx

Client

Install Caddy : apt install -y caddy ~

after set dns)

curl bot --nginx -d wd 22.c (an) (ip)

ngt

yes

No

Congrate etc/nginx / etc-etc

system related nginx

nginx.conf → web server → serve static (Passet Client)
reverse proxy → proxy server
15-28 (fahay proxy - client)

Debug : docker ps
docker logs --tail 100 -f strapi # Docker view

shutdown -h now

Admin 10.10.10.10 → 10.10.10.10
Docker Pods

Nginx.conf เพิ่ม |users-permissions|

Visual host (ยาก)

(สำหรับ Deploy จริง)

docker compose up -d --build

