

Cài đặt môi trường OS - SSL - Domain

Lưu ý:

- Thực hiện cài đặt môi trường trước, sau đó mới tiến hành deploy 2 container CMND. Quá trình cài đặt cần server được kết nối internet.
- Cài đặt cho server **nvidia driver phiên bản: 530.30.02**

link: <https://www.nvidia.com/download/driverResults.aspx/199985/en-us/>

1. Các file setup đính kèm (file envsetup.zip)

- **envDeUbu.sh** dùng để cài đặt các packages phụ thuộc của API ID. Các packages bao gồm:

- curl
 - wget
 - docker (phiên bản 24.0.2)
 - zip
 - nginx (SSL-Domain, phiên bản 1.22.1)
- **deploy-nginx.sh** dùng để cài đặt nginx container.
- **logshell**: thư viện logging của bash.
- **nginxgenconfig.toml**: file cấu hình của nginx.

2. Cài đặt môi trường, deploy Nginx (SSL+Domain)

- **Điều kiện cài đặt:**

- Server sử dụng hệ điều hành Linux. (Ubuntu, Centos,..)
- Server đã được kết nối Internet.
- Nếu server sử dụng Red Hat OS, đảm bảo server đã được subscribe và register.
- Đã có domain đã đăng ký và file SSL.

Step1.

- Thay tên **domain** đã đăng ký cho server vào mục VHOST (dòng 31) của file **nginxgenconfig.toml** (có thể dùng kết hợp giữa domain và IP local/public)
ví dụ: VHOST = '10.6.30.250 domain1.com domain2.com.vn'
- Thay tên **certificate** file sử dụng (tên của cặp file .pem (.crt), .key) vào mục CERNAM (dòng 32) của file **nginxgenconfig.toml**
ví dụ: CERNAM = 'cert'
ứng với 2 file cert.key và cert.crt

```

17 [[Containers]]
18 Name = "all"
19 State = {Running = true}
20 [Containers.env]
21 VPROTO = "http"
22 HTTP_METHOD = 'noredirect'
23 VHOST = '192.168.1.30 domain1.com.vn domain2.com.vn' # name of domain you use, or IP if use local
24 CERNAME = 'certificate-name' # name of certificate (example: certificate-name.key, certificate-name.crt(.pem))
25 DISABLEGENUPSTREAM = 'true'
26 VPORT = '80'
27 VHOSTOPT = 'include statuslocal.conf;'
28 VPATHDESTS = ""/idfull/,UN:idsmartocrvn
29 |/hello/,UN:hello""
30

```

Step2.

- Chuyển sang phiên làm việc của root với lệnh:
su -

Step3.

- Dùng lệnh **cd** để chuyển thư mục làm việc tới thư mục cài đặt **Os**
- Cấp quyền execute cho file **envDeUbu.sh** bằng lệnh:
chmod +x ./envDeUbu.sh
- Chạy lệnh sau để bắt đầu quá trình cài đặt.
./envDeUbu.sh

Step4.

- Copy cặp certificate file đã chuẩn bị (cặp file .pem (.crt), .key) vào folder
/mnt/containerdata/nginxgen/etc/nginx_certs
- Sau đó chạy lệnh để apply thay đổi:
docker restart nginxgen

Step5: . Kiểm tra kết quả cài đặt:

- **Cách 1:** Chạy lệnh:
curl -I -k -X GET https://your_domain_name/hello
hoặc:
curl -I -k -X GET http://ip-lan/hello
(với ip-lan là IP trong mạng Lan nội bộ)
Kết quả thành công với lệnh curl (HTTP/2 200).

```

HTTP/2 200
server: nginx
date: Thu, 22 Jun 2023 04:57:17 GMT
content-type: text/html
expires: Thu, 22 Jun 2023 04:57:16 GMT
cache-control: no-cache
x-content-type-options: nosniff
x-robots-tag: none
x-permitted-cross-domain-policies: none
x-request-id: bbd2c3a69cb66f6ccdfdb2825b10988
x-app-request-id: bbd2c3a69cb66f6ccdfdb2825b10988
x-frame-options: SAMEORIGIN
strict-transport-security: max-age=31536000

```

Quá trình cài đặt sẽ triển khai một container là "hello" để test nginx, sau bước này có thể xóa container này với lệnh:

docker rm -f hello

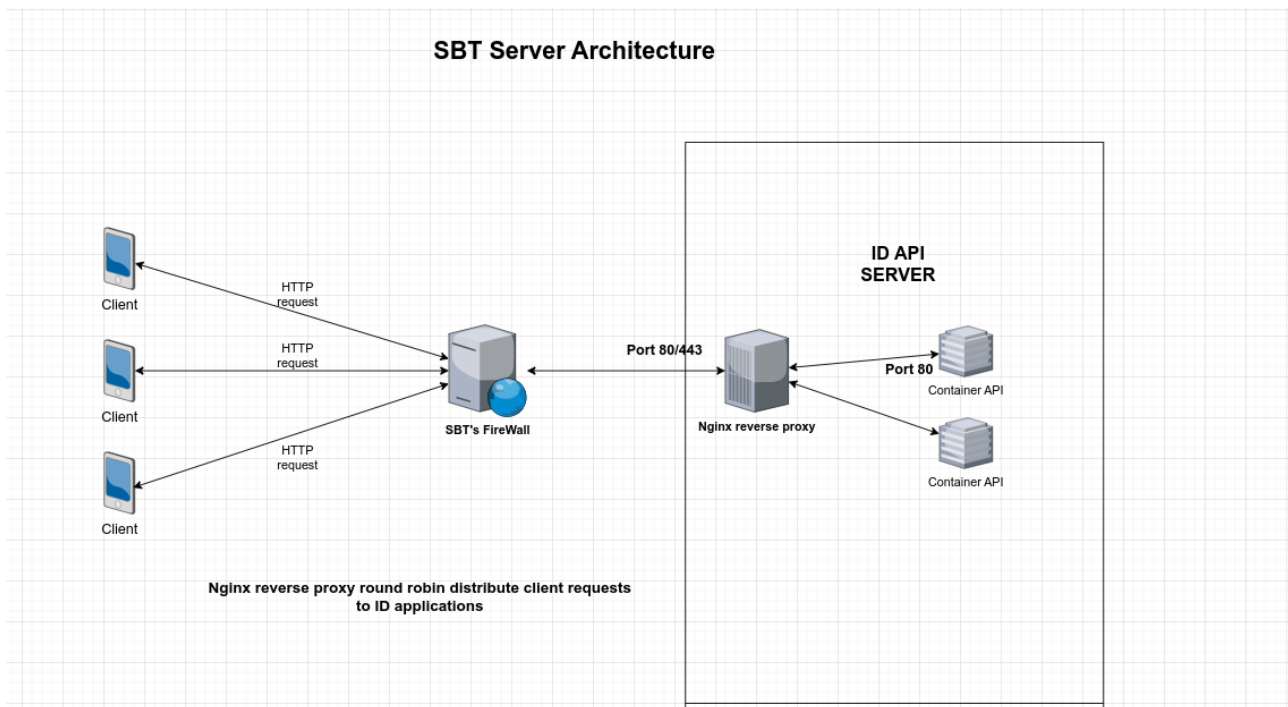
- **Cách 2:** Hoặc truy cập trang web: https://your_domain_name/hello. (Với [your_domain_name](#) là tên đã đăng ký cho server) hoặc <http://ip-lan/hello> với ip-lan là IP trong mạng Lan nội bộ. Nếu trang web xuất hiện như dưới đây thì tức là đã cài đặt thành công SSL và domain.

NGINX

Server address:	172.172.0.3:80
Server name:	hello
Date:	22/Jun/2023:02:33:51 +0000
URI:	/hello/

- Log của quá trình cài đặt được lưu tại:
/tmp/log-deploy.log

3. Cấu trúc hệ thống với nginx reverse proxy



- Container nginx-proxy đóng vai trò reverse proxy, thực hiện các chức năng:
 - Cấu hình domain, cung cấp lớp bảo mật SSL. Nginx sẽ cung cấp dịch vụ trên port 443 của server deploy.
 - Phân tải giữa các container ID.
- Các request từ client qua FW của SBT sẽ tới nginx (nginx listen port 80/443). Tại đây các request được nginx phân chia đều (round robin) cho các container ID để xử lý (container face listen port 80).
- Response message trả về từ API của Face Searching sẽ được redirect lại cho nginx sau đó nginx sẽ trả lại kết quả này cho client gửi request.

4. Deploy ID container.

- Deploy theo tài liệu triển khai Face Searching đi kèm.
Quan trọng: Deploy lần lượt các container theo hướng dẫn với tên id0, id1 (tên container là quan trọng, vì nginx sẽ nhận diện các container theo tên).
- Sau khi deploy thành công, thực hiện test các chức năng API được bàn giao với Postman.

https://your_domain_name/idfull/v1/recognition

- Check app_log của từng container để xác nhận request đã được forward đều tới 2 container.