**CÁC BƯỚC TRIỂN KHAI NODEJS LÊN VPS CENTOS**

**Một số phần mềm được sử dụng tới trong loạt bài này:**

1. [**WinSCP**](https://winscp.net/eng/download.php) **- hỗ trợ trình duyệt file của Centos**
2. [**Putty**](https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html) **- command line**
3. [**Atom**](https://atom.io/) **- trình soạn code**
4. [**NoSQLBooster for MongoDB**](https://nosqlbooster.com/downloads) **- quản lý csdl mongodb**

**MENU**

1. ***Cài đặt cơ bản***
2. ***Cài đặt các nền tảng cần thiết để chạy NodeJS***
3. ***Trỏ tên miền VPS***
4. ***Clone ứng dụng bằng git***
5. ***Cài đặt nginx và thiết lập web block (virtual host)***
6. ***Tạo chứng chỉ SSL (free) bằng Let’s Crypt***

**BẮT ĐẦU**

1. **Cài đặt cơ bản**

**# Mặc định VPS đã cài sẵn tường lửa (Firewall)**

* **Cập nhật hệ thống: yum update -y**
* **Cập nhật lại ngày giờ theo giờ địa phương:**

**rm -f /etc/localtime #câu lệnh thứ nhất**

**ln -s /usr/share/zoneinfo/Asia/Ho\_Chi\_Minh /etc/localtime #câu lệnh thứ 2**

* **Đồng bộ Network Time Protocol NTP về server Việt Nam:**
* **Cài đặt NTP: yum install -y ntpdate**
* **Đồng bộ NTP server Việt Nam: ntpdate vn.pool.ntp.org**

1. **Cài đặt các nền tảng cần thiết**

* **Cài đặt git để clone ứng dụng từ github về: sudo yum install git -y**
* **Cài đặt Nodejs & npm:**
* **Cài NVM (Node Version Manager) bằng 2 lệnh sau:**

**curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.8/install.sh | bash #Câu lệnh thứ nhất**

**source /root/.bash\_profile #Câu lệnh thứ 2**

* **Cài đặt NodeJs:**

**nvm list-remote** # hiển thị danh sách version của nodejs

**nvm install v14.5.0 -y** #chọn version muốn cài từ list trên

* **Cài đặt pm2 để chạy ngầm ứng dụng NodeJS trên VPS: npm install pm2 -g**
* **Cài đặt mongodb:**
* **Tạo 1 repository: sudo vi /etc/yum.repos.d/mongodb-org.repo**
* **Cấu hình cho mongodb bằng cách insert đoạn sau vào file đang mở ở trên:**

**[mongodb-org-4.2]**

**name=MongoDB Repository**

**baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/4.2/x86\_64/**

**gpgcheck=1**

**enabled=1**

**gpgkey=**[**https://www.mongodb.org/static/pgp/server-4.2.asc**](https://www.mongodb.org/static/pgp/server-4.2.asc)

**#Dành cho các bạn không quen dùng vi để soạn thảo văn bản trên Centos**

**#Muốn chèn dữ liệu thì ta bấm phím i**

**#Để thoát khỏi chế độ soạn dữ liệu, bấm phím ESC**

**#Muốn lưu dữ liệu ta gõ :wq**

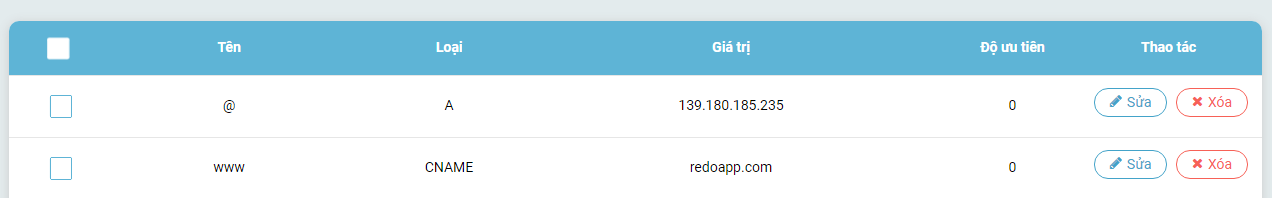
**#Muốn thoát mà không lưu ta gõ :q!**

**#Các bạn lưu lại file cấu hình trên bằng nano**

* **Cài đặt mongodb: sudo yum install -y mongodb-org -y**
* **Khởi động dịch vụ mongodb: sudo systemctl start mongod**
* **Thiết lập mongodb chạy cùng vps: sudo systemctl enable mongod**

1. **Trỏ tên miền về VPS**

**Để trỏ được domain về VPS ( cũng như hosting ) chúng ta chỉ cần thiết lập 2 thông số như hình:**

****

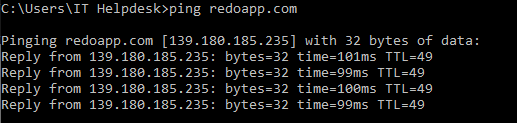
**Lưu ý: Các thông số trong cột “Giá trị” ( Value) là IP VPS và domain của bạn!**

**Cấu hình xong bạn có thể ping thử xem domain của mình đã hoạt động bình thường chưa bằng câu lệnh:**

**(Nhấn tổ hợp phím *Windows + R* để mở hộp thoại *Run)***

**ping redoapp.com #các bạn thay thế redoapp.com bằng domain của các bạn**

**Kết quả như sau là ok:**

****

**Phần IP đã khớp với Ip được cấu hình trong domain**

**#Trong trường hợp trước đó domain của bạn đang trỏ tới 1 host khác. Bạn ping nó vẫn nhận địa chỉ IP cũ thì dùng lệnh:** ipconfig /flushdns **rồi Enter** **để xóa cache nhé!**

1. **Clone dự án về VPS bằng git:**

**# Có thể cd vào thư mục sẵn có trên VPS, chẳng hạn: cd /home**

**# Hoặc có thể tạo mới 1 thư mục trên VPS, chẳng hạn: mkdir -p /apps**

**#Sau đó cd vào thư mục vừa tạo: cd /apps**

1. **Clone dự án đã có sẵn được upload lên github.com về VPS**

**Dùng lệnh: git clone <link\_project\_trên\_github>**

1. **Vào thư mục của project vừa clone ở trên: cd <ten\_thu\_muc\_cua\_project>**

**Chạy lệnh: npm install để cài đặt các gói(packet) trong project NodeJs**

**V. Cài đặt NGINX và thiết lập WEB BLOCK**

1. **Tạo 1 Nginx respository: sudo yum install epel-release -y**
2. **Cài đặt nginx: sudo yum install nginx -y**
3. **Chạy nginx: sudo systemctl start nginx**
4. **Thiết lập nginx chạy tự động cùng với VPS: sudo systemctl enable nginx**
5. **Nếu bạn đang dùng tường lửa thì cần add dịch vụ http và https:**

**sudo firewall-cmd --permanent --zone=public --add-service=http #add http**

**sudo firewall-cmd --permanent --zone=public --add-service=https #add https**

**sudo firewall-cmd --reload #khởi động lại dịch vụ tường lửa**

1. **Test xem nginx đã chạy ok chưa:**

**Các câu lệnh liên quan tới nginx:**

**service nginx start # khởi động dịch vụ nginx**

**service nginx stop # dừng nginx**

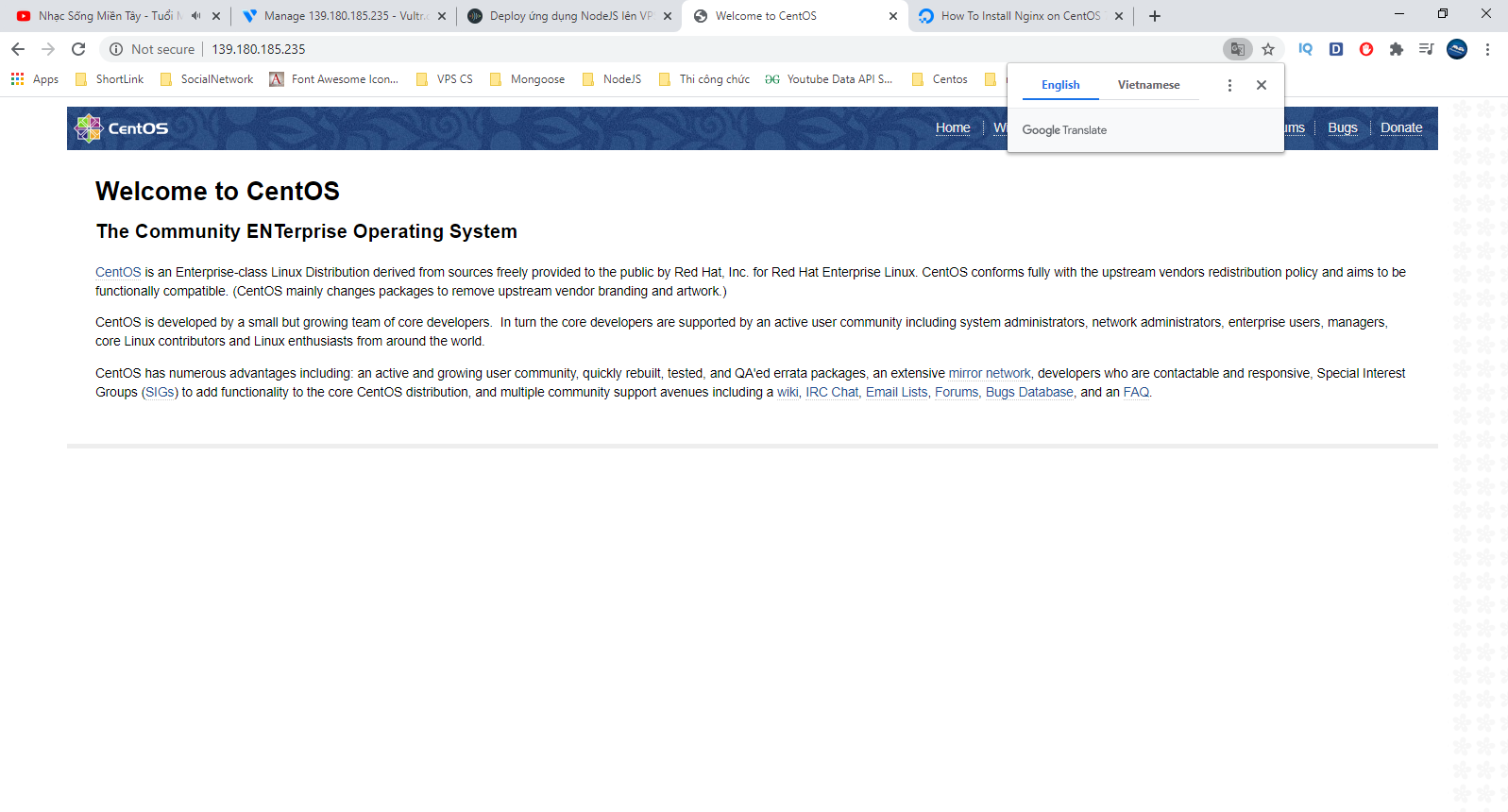
**service nginx restart # khởi động lại nginx**

**sudo nginx -t # kiểm tra cấu hình nginx**

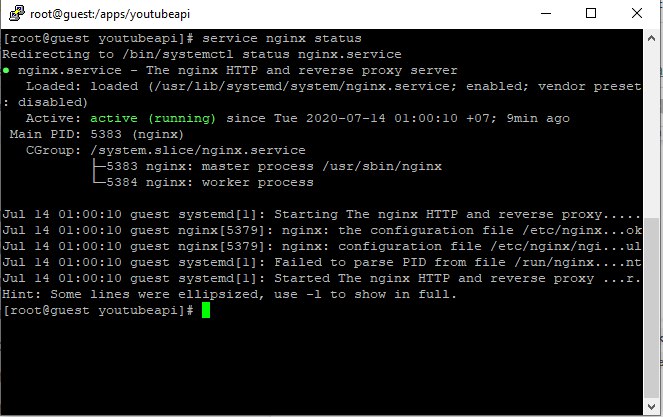
**service nginx status # kiểm tra trạng thái của nginx**

**Test nginx trên trình duyệt:**

[**http://server\_domain\_name\_or\_IP/**](http://server_domain_name_or_ip/) **#**[**server\_domain\_name\_or\_IP**](http://server_domain_name_or_ip/)**: địa chỉ VPS của bạn**

****

**Nếu kết quả ra được như này là ok. Còn không thì kiểm tra lại trạng thái của nginx xem đã khởi động chưa bằng câu lệnh: service nginx status sẽ cho ra màn hình tương tự sau**

****

**Nếu kết quả ra như hình trên (active) nhưng vẫn không được thì xem lại các câu lệnh add dịch vụ vào tường lửa ở phía trên nhé**

1. **Tạo Web Block (Virtual Host)**

***Tại sao lại phải tạo web block?***

**Mặc định nginx trỏ về thư mục /usr/share/nginx/html**

**Nếu bạn chỉ có 1 website upload lên VPS thì bạn có thể “nhét” chúng vào thư mục trên. Tuy nhiên, nếu bạn có nhiều hơn 1 project muốn upload lên VPS hoặc đơn giản là bạn không thích cái đường dẫn thư mục mặc định loằng ngoằng của Nginx. Bạn muốn customize lại, chẳng hạn như thư mục /apps hoặc /home như ở phần III ở trên thì chúng ta cần biết tới web block.**

**Lưu ý: Tạo web block trong trường hợp các bạn đã có sẵn 1 domain ( tên miền ) và muốn trỏ nó về thư mục mà chúng ta đã clone project ở phần III nhé!**

**Cụ thể, ở bước III, mình đã tạo 1 thư mục tên là apps. Sau đó clone project vào. project mình sau khi clone về nằm trong thư mục /apps/youtubepi(youtubeapi là thư mục được tạo ra sau khi clone project). Và mình muốn trỏ tên miền của mình là redoapp.com về thư mục này.**

* **Gán quyền cho nginx có quyền truy xuất thư mục project của mình**

**chown -R nginx:nginx /apps/youtubeapi**

* **Tạo 1 file config để thiết lập 1 virtual host:**

**nano /etc/nginx/conf.d/redoapp.com.conf**

* **Chèn nội dung như sau vào file redoapp.com.conf đang được mở ở phía trên:**

**server {**

**listen 80;#port mặc định của nginx**

**server\_name redoapp.com** [**www.redoapp.com**](http://www.redoapp.com)**; #domain của bạn**

**location / {**

**proxy\_pass** [**http://localhost:5000**](http://localhost:5000)**; #port của project nodejs**

**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;**

**}**

**}**

**#Dành cho các bạn không quen xài trình soạn thảo nano của Centos**

**#Sau khi mở(tạo file), chúng ta chỉ cần chuột phải để paste nội dung đã copy**

**# Ctr + O để ghi tiến hành ghi file**

**# Enter để submit**

**# Ctr+X để thoát khỏi nano**

* **Test xem config có lỗi lầm gì không: sudo nginx -t**

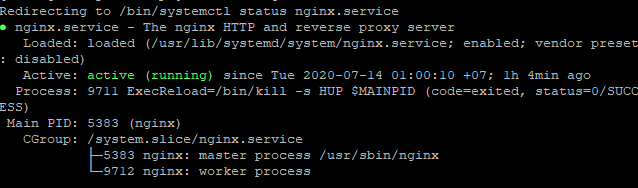
**# Kết quả cấu hình thành công sẽ cho đầu ra như sau:**

**nginx: the configuration file /etc/nginx/nginx.conf syntax is ok**

**nginx: configuration file /etc/nginx/nginx.conf test is successful**

* **Khởi động lại nginx: service nginx reload**
* **Kiểm tra lại trạng thái của nginx: service nginx status**

**#Kết quả được như dưới hình là ok**

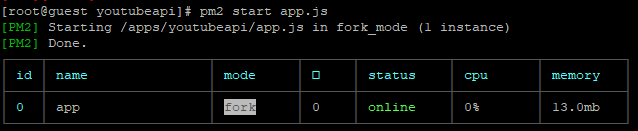
****

**#Vì chúng ta trỏ về project nodejs nên đầu tiên chúng ta cần khởi động project lên**

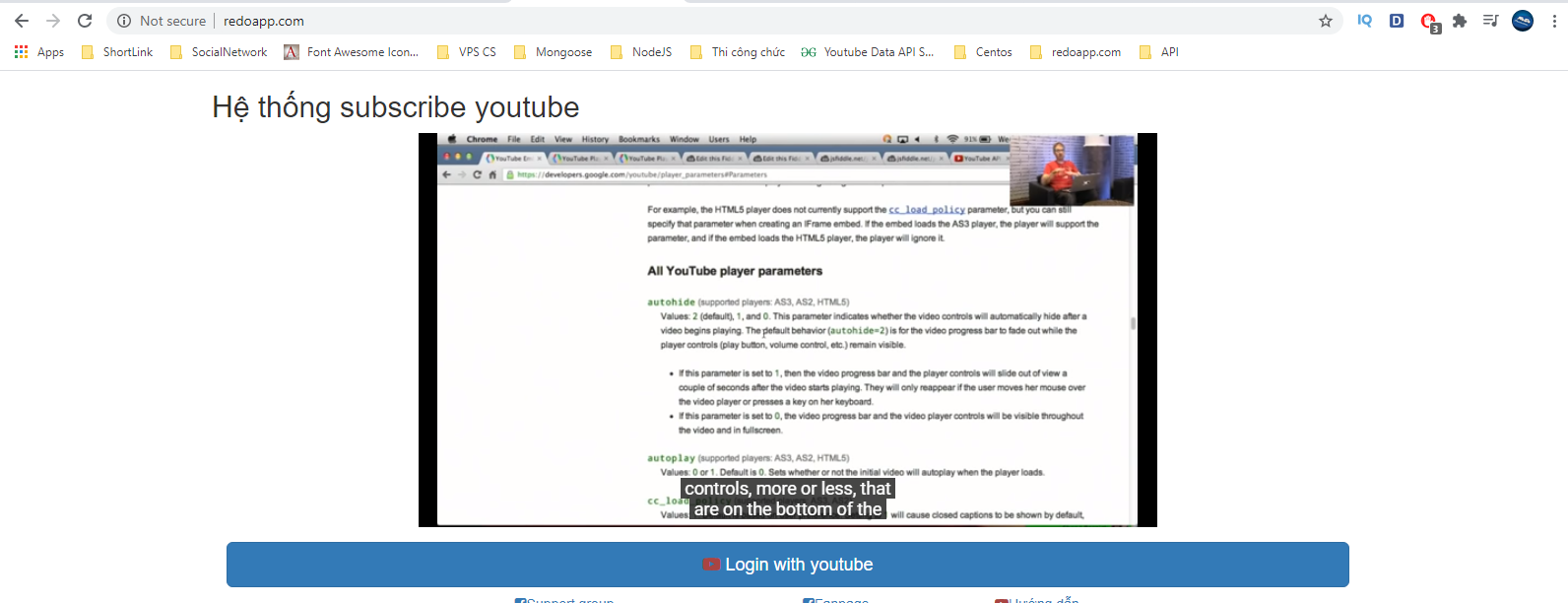
**## Bước 1: vào thư mục project của mình: cd /apps/youtubeapi**

**## Bước 2: khởi động nodejs: pm2 start app.js #app.js là trang chính project của mình.**

**## Kết quả thu được như sau:**

****

* **Test kết quả sau khi cấu hình web block trên trình duyệt:**

****

**Một số lưu ý: Nếu bạn gặp lỗi 502 Bad Gateway hoặc 403 Forbidden**

**thì các bạn có thể chạy câu lệnh: setenforce permissive ( áp dụng cho** [**Centos SELinux**](https://vinasupport.com/selinux-la-gi-cach-vo-hieu-hoa-selinux-tren-centos/) **nhé)**

**VI. Tạo chứng chỉ** [**SSL**](https://vhost.vn/chung-chi-ssl/ssl-la-gi/) **free với Let’s Crypt**

**Trang không có SSL sẽ có dạng như sau:**

****

**Tên miền chưa được chứng thực sẽ có dạng:** [**http://redoapp.com**](http://redoapp.com)

**Tên miền được chứng thực sẽ có dạng: https://redoapp.com**

**Có nhiều loại SSL khác nhau nhưng trong tutorial này mình sẽ hướng dẫn các bạn lấy SSL free và tự động gia hạn khi SSL bị hết ( 03 tháng ).**

1. **Tạo domain mail:**

**Để chứng thực được SSL chúng ta cần có mail domain để chứng tỏ rằng domain mà bạn muốn chứng thực thuộc quyền sở hữu của bạn:**

**Việc tạo mail cũng khá dễ, chúng ta chỉ cần thực hiện theo các bước ở** [**link này**](https://support.tenten.vn/449881-H%C6%B0%E1%BB%9Bng-D%E1%BA%ABn-T%E1%BA%A1o-Email-Google-Theo-T%C3%AAn-Mi%E1%BB%81n)

1. **Cài đặt SSL**

* **Kích hoạt truy cập đến kho EPEL trên server của bạn bằng lệnh: sudo yum install epel-release**
* **Cài đặt certbot-nginx: sudo yum install certbot-nginx -y**
* **Add port 80 và 443:**

**sudo iptables -I INPUT -p tcp -m tcp --dport 80 -j ACCEPT #add port 80**

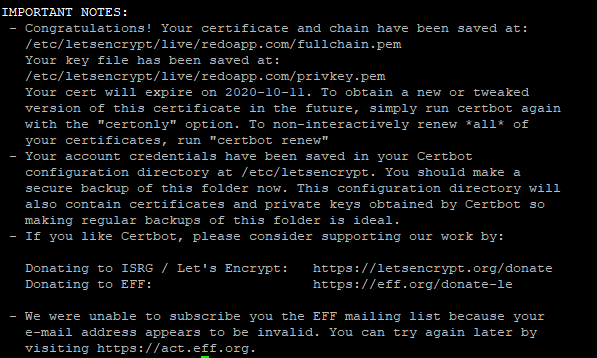
**sudo iptables -I INPUT -p tcp -m tcp --dport 443 -j ACCEPT #add port 443**

* **Lấy chứng chỉ: sudo certbot --nginx -d redoapp.com -d www.redoapp.com**

**#Các bạn thay thế redoapp.com bằng tên miền (domain) của các bạn nhé!**

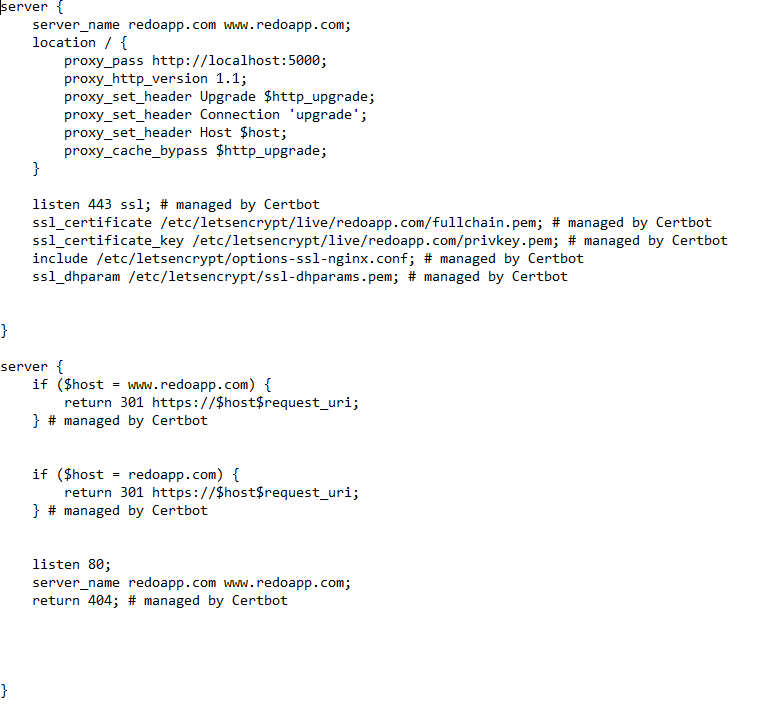
**Nó sẽ hỏi nhập domain email . Bạn cần cung cấp email mình đã đăng ký ở trên và chấp thuận (A) và đồng ý (Y) ở vài câu hỏi của nó.**

**Cuối cùng chúng ta có kết quả như sau:**

****

**#Mở lại file config block ở phần V, chúng ta sẽ thấy nó khác với lúc đầu chúng ta config. Đó là do certbot-nginx đã tự động thêm vào**

**#Nội dung như sau(phần này mình nói thêm để các bạn hiểu chứ không cần thiết):**

****

**Như vậy chúng ta đã đăng ký chứng thực thành công. SSL đã tự động được thêm vào địa chỉ web (domain) của chúng ta. Giờ đây, bên cạnh domain của chúng ta đã có biểu tượng ổ khóa và tên miền của chúng ta có dạng:** [**https://redoapp.com**](https://redoapp.com)

****