FCA INSTALL

https://github.com/rupesx26/fca-aws

Create Ec2 Instance

Connect with SSH using putty

Login as a root user - sudo su

If you want to switch back to normal user use command - sudo su ec2-user

sudo apt-get update

=======================================================

Install Node VIA NVM so that you can manage versions

curl https://raw.githubusercontent.com/creationix/nvm/master/install.sh | bash

source ~/.bashrc

nvm ls-remote

nvm install v12.16.3 (use the version number)

nvm list (to list available node version)

nvm use v12.16.3 (to select the default version)

node --version (verify node version)

=================================================

Using Github, BitBucket or Gitlab

sudo apt-get install -y git

git --version (verify git version)

git clone https://github.com/rupesx26/fca-aws

=========================

INSTALLING NODE DEPENDENCIES AND STARTING NODE SERVER

ls

cd fca-aws

npm install (inside the website directory)

npm run build (create a build for production)

npm run start (to start the production server)

PLEASE NOTE : Open PORT 3000 in AWS security group inbound rules - CUSTOM TCP 3000

=========================================

INSTALLING PM2 - Process manager to keep alive the node server after closing the session also -https://pm2.keymetrics.io/docs/usage/quick-start/

killall -9 node (this will kill all the node process)

npm install pm2 -g (install it globally)

pm2 start scripts/startProd.js (this will run the node app even if you close the terminal)

pm2 list (to check the running process)

======================================================

pm2 startup (For automatically running PM2 when the server stop/restarts)

This will output a script in the terminal which you need to run, copy the script that it outputs and run it in the terminal window. Now, save all the currently running processes so that they can be run again whenever PM2 restarts either manually or by a script with the following command:

pm2 save (to save the config)

======================================================

To remove port no 3000 - we should route 3000 to port 80 - Now lets install NGINX SERVER for that - FOR LINUX ITS DIFFERENT

sudo apt-get update

nginx -v

sudo service nginx start (use sudo service nginx (then command))

service nginx status

CHECK THE URL OF Ec2 it will show NGINX DEFAULT PAGE

PLEASE NOTE : Open PORT 80 in AWS security froup inbound rules - HTTP 3000

NOW OPEN THE NGIX config file and do the below setting (https://www.cyberciti.biz/faq/nginx-linux-restart/, https://www.tecmint.com/nginx-as-reverse-proxy-for-nodejs-app/)

cd /etc/nginx/sites-available

systemctl restart nginx

sudo nano /etc/nginx/nginx.conf

===

server {

listen 80;

server\_name 13.127.255.36;

location / {

proxy\_set\_header X-Forwarded-For $remote\_addr;

proxy\_set\_header Host $http\_host;

proxy\_pass http://13.233.237.79/:3000;

}

}

}

===

sudo apt-get remove nginx

sudo systemctl restart nginx (restart nginx)

sudo systemctl status nginx (see nginx status)

sudo systemctl enable nginx (always on after reboot also)

3.7.139.127

sudo nginx -t

=========================

ATTACHING DOMAIN TO EC2

https://aws.amazon.com/getting-started/hands-on/get-a-domain/

Create a Route 53

Add Domain

Change NAmserver at domain end

Add A record with www

Add A record with Alais as www

Edit NGINX CONFIG

sudo nano /etc/nginx/nginx.conf

===

server {

listen 80;

server\_name gocatchy.com;

location / {

proxy\_set\_header X-Forwarded-For $remote\_addr;

proxy\_set\_header Host $http\_host;

proxy\_pass http://3.7.153.66:3000;

}

}

}

https://medium.com/@Keithweaver\_/setting-up-mern-stack-on-aws-ec2-6dc599be4737

https://www.serverlab.ca/tutorials/linux/web-servers-linux/how-to-configure-multiple-domains-with-nginx-on-ubuntu/

gocatchy.com

https://www.digitalocean.com/community/tutorials/how-to-set-up-nginx-server-blocks-virtual-hosts-on-ubuntu-14-04-lts

https://www.serverlab.ca/tutorials/linux/web-servers-linux/how-to-configure-multiple-domains-with-nginx-on-ubuntu/

PERFECT :https://www.youtube.com/watch?v=ehITvx8VPFI

=================

SSL

https://certbot.eff.org/lets-encrypt/ubuntuxenial-nginx

certbot -d gocatchy.com -d www.gocatchy.com

Enter all details and new certficate will be obtained

Copy the path and details that comes on screen

- Congratulations! Your certificate and chain have been saved at:

/etc/letsencrypt/live/gocatchy.com/fullchain.pem

Your key file has been saved at:

/etc/letsencrypt/live/gocatchy.com/privkey.pem

Your cert will expire on 2020-08-11. To obtain a new or tweaked

version of this certificate in the future, simply run certbot again

with the "certonly" option. To non-interactively renew \*all\* of

your certificates, run "certbot renew"

- Your account credentials have been saved in your Certbot

configuration directory at /etc/letsencrypt. You should make a

secure backup of this folder now. This configuration directory will

also contain certificates and private keys obtained by Certbot so

making regular backups of this folder is ideal.

GO TO THIS FOLDER /etc/letsencrypt/live/gocatchy.com/ - here all your certficate and pem files are stored

Navigate to the etc/nginx/sites-available folder and edit the default conf

ANd paste the below

https://certbot.eff.org/lets-encrypt/ubuntuxenial-nginx

server {

listen 80;

server\_name gocatchy.com www.gocatchy.com;

return 301 https://$host$request\_uri;

}

server {

# SSL configuration

listen 443 ssl http2;

listen [::]:443 ssl http2;

server\_name gocatchy.com;

return 301 https://www.$server\_name$request\_uri;

}

server {

listen 443 ssl http2 default\_server;

listen [::]:443 ssl http2 default\_server;

ssl\_certificate /etc/letsencrypt/live/gocatchy.com/fullchain.pem; # managed by Certbot

ssl\_certificate\_key /etc/letsencrypt/live/gocatchy.com/privkey.pem; # managed by Certbot

ssl\_protocols TLSv1 TLSv1.1 TLSv1.2;

ssl\_prefer\_server\_ciphers on;

ssl\_ciphers 'EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH';

location / {

proxy\_pass http://172.31.5.235:3000;

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;

}

}

==========

npm i forever -g

forever start scripts/startProd.js

forever stop scripts/startProd.js

How to test redirect non-www to www over SSL with Nginx serer and curl

The syntax is as follows for the curl command:

curl -IL https://theos.in/

curl -IL http://theos.in/

http://gocatchy.com

http://www.gocatchy.com

https://gocatchy.com

https://www.gocatchy.com

sudo systemctl restart nginx (restart nginx)

sudo systemctl status nginx (see nginx status)

sudo systemctl enable nginx (always on after reboot also)

cd etc/nginx/sites-available/

curl -I https://www.gocatchy.com

forever stop scripts/startProd.js

IMPORTNAT NOTE : whenever editing code in the file we have to stop and start the node or forever or pm2 process again

==========================

Client ID: 1088396227869-e0lhb2ooes4i0h0g1n6tnpbuf82qo9o5.apps.googleusercontent.com

Client Secret: 1DP0yosxcMfjkUJnM2omi5Mh

Refresh Token: 1//04aRNbE1TV6exCgYIARAAGAQSNwF-L9IrGpaIGBD6QRm33uGaNV-yQD0hchWT76I8\_Hy\_w8TPBMZxw-01cZGnnjJP6dPy3lIka9M

Access Token: ya29.a0AfH6SMCRzWgBJbzquBjUAvP2mlT7m7554jDQfcwUw-849uEN7SiwDcZWzfVllHiqJLjkkgdRICKzfcn277yn5EBNPC4bTyb-bjTLfYxMMuBMCb-P4zN0uxt85KO78N8yQYs5d4-N8yQtZbcslq3cR89V3ZCIKSyeTeg

==========================

https://stackoverflow.com/questions/51933601/what-is-the-definitive-way-to-use-gmail-with-oauth-and-nodemailer

https://medium.com/@nickroach\_50526/sending-emails-with-node-js-using-smtp-gmail-and-oauth2-316fe9c790a1

host: 'smtp.gmail.com',

port: 465,

secure: true,

auth: {

type: 'OAuth2',

user: "support@makemelive.in",

clientId: "1088396227869-e0lhb2ooes4i0h0g1n6tnpbuf82qo9o5.apps.googleusercontent.com",

clientSecret: "1DP0yosxcMfjkUJnM2omi5Mh",

refreshToken: "1//04aRNbE1TV6exCgYIARAAGAQSNwF-L9IrGpaIGBD6QRm33uGaNV-yQD0hchWT76I8\_Hy\_w8TPBMZxw-01cZGnnjJP6dPy3lIka9M",

accessToken: local\_settings.my\_oauth\_access\_token

}

user: "support@makemelive.in",

clientId: "1088396227869-e0lhb2ooes4i0h0g1n6tnpbuf82qo9o5.apps.googleusercontent.com",

clientSecret: "1DP0yosxcMfjkUJnM2omi5Mh",

refreshToken: "1//04aRNbE1TV6exCgYIARAAGAQSNwF-L9IrGpaIGBD6QRm33uGaNV-yQD0hchWT76I8\_Hy\_w8TPBMZxw-01cZGnnjJP6dPy3lIka9M",