

## Base Image:

```
docker load < /opt/eltropy/docker/ubuntu-18.04-eltropy.tar
docker tag 55fb04f6e94f eltropy-new-img
docker run -p 27017:27017 -p 5672:5672 -p 15672:15672 -p 6379:6379 -p 5432:5432 -p
1883:1883 -p 15675:15675 -v /tmp:/elhost -it --name eltropy-new eltropy-new-img /bin/bash (We
need to pull image only once)
```

In docker desktop under settings:

- give access to location: /opt/eltropy

- Change cpu/memory configuration and apply and restart. (7gb ram, 5cpu)

- Apply and restart.

- Re run eltropy-new-image

In terminal

```
docker start eltropy-new
```

```
docker attach eltropy-new
```

```
sudo service supervisor stop
```

```
sudo service supervisor start
```

```
supervisorctl status
```

Also ref: <https://gist.github.com/abhishek-eltropy/31f3d176a3daf4ab1e932dec3537b0c2>

## Pull repo using Github desktop

Radiant backend:

In /opt/eltropy/repo/radiant-backend/ pull <https://github.com/eltropy/radiant-backend>

Goremon:

/opt/eltropy/repo/goremon/ pull <https://github.com/eltropy/goremon>

Radiant-gitops:

/opt/eltropy/repo/radiant-gitops pull <https://github.com/eltropy/radiant-gitops>

## Setup Radiant Backend

<https://radiant.eltropy.com/docs/getting-started/set-up-local-system/>

Run second docker Image:

```
docker-compose -f
```

```
/opt/eltropy/repo/radiant-backend/local-dev-setup/setup-dev-env-compose.yaml up
```

Run Envoy: `envoy -c /opt/eltropy/repo/radiant-gitops/local/config/envoy/envoy.yaml`

*Check DB's in respective tools:*

For postgres open DBeaver set up new postgres connection and use:

Database      eltrophydb  
Username      eltrophy  
Password      swhad0o9

For Mongo open robo3t, add new a connection if not there already, in authentication tab use:

Database: admin  
Username:"eltropy  
Password: eltrophy1234#

DB Migration: (NEW)

*Installation:*

`cd /opt/eltropy/repo/radiant-backend/java/`

Install Amazon Corretto 17 (JAVA) from self service and maven version is 3.9.6 in tools folder as mentioned in this doc: <https://radiant.eltropy.com/docs/java/setup-local-system/> (add path in zshrc)

*Running db migration:*

In Common-Libs, grpc-protobuf and root Java level, run: `mvn clean install`

(For any errors check if everything is fine in consul, indentation error in-aws and conf file probably)

Finally,

`cd` in `java/db-migration-app` and run:

`mvn spring-boot:run` (You just need to run this command everytime if there are changes)

Run backend apps:

`cd /opt/eltropy/repo/radiant-backend/golang/<APP NAME>/internal/cmd/rest`

TO run : `go run <APP NAME>`

#Radiant apps:

1. Auth-app
2. domain-app
3. Cms-app

4. gatewayapp grpc not rest
5. Message-consumer-app
6. Messagesapp
7. Settings-app
8. admin-portal
9. cd /opt/eltropy/go/bin (config-app decrypt/encrypt "")

## Run Frontend apps:

cd /opt/eltropy/repo/goremon/

Check nvm version, if not install

Brew install nvm

Add this in zshrc file:

source \$(brew --prefix nvm)/nvm.sh

source ~/.zshrc

Install node if not already:

nvm install 10.22.1 --default

nvm install 16.16 (16.16 is required for omnicom only)

nvm use default

#Goremon apps:

Auth : npm i && npm start

Cms-app npm i && npm run start:local

Main-app npm i && npm run start:local

Messages export NODE\_OPTIONS="--max-old-space-size=8192" && npm i && npm start

Omnicom nvm use 16.16.0 && npm i && npm start

Admin-portal npm i && npm start

Lobby npm i && npm start #not required now

Settings npm i && npm start

Verification (not needed)

/messages/mycu and bridge

Analytics, content management app for activity log

## HTTPS SETUP - Install mkcert

brew install mkcert

brew install nss

mkcert -install

cd /opt/eltropy/conf

```
mkcert "*.local-eltropy.com" "*.localtropy.com" localtropy.com local-eltropy.com localhost
127.0.0.1 ::1
mv _wildcard.local-eltropy.com+6.pem mkcert.pem
mv _wildcard.local-eltropy.com+6-key.pem mkcert.pem.key
```

## Ngrok

Do the steps in the Ngrok setup section here:

<https://docs.google.com/document/d/1S6vihYHYuqdjZeLAHt8nn6NauoGiw-vw/edit> (basically brew install ngrok/ngrok/ngrok then sign up and add token)

After installing, signup, and running the ngrok config cmd, to start the tunnel run the following:  
ngrok http demo.localtropy.com:443

You will get an output like this:

Forwarding                https://fa87-103-240-205-71.in.ngrok.io -> http://demo.localtropy.com:80

In consul local/config/common/integrations, modify voiceflowapihosturl to use above URL i.e:  
voiceflowapihosturl: "https://fa87-103-240-205-71.in.ngrok.io "

Also run the following cmd in redis with above url:

```
redis-cli -a eltropy@1749560068
```

```
set hostNameToEIDomainIdCache:<url> 72430
```

For example:

```
set hostNameToEIDomainIdCache:fa87-103-240-205-71.ngrok-free.app:443 72430
```

```
set hostNameToEIDomainIdCache:fa87-103-240-205-71.ngrok-free.app 72430
```

## Encryption using config app

<https://radiant.eltropy.com/docs/configuration/config-app-command-line-utility/#encrypt-consul-config-key>

The encrypted string is the secret\_key and nonce is secret\_key\_nonce

This is the method to encrypt any key, remember this for later:

```
cd /opt/eltropy/go/bin
```

```
config-appencrypt stringtoencrypt
```

## EMQX Setup (Message broker)

<https://docs.google.com/document/d/1MKA-VubOGYF8qP8kx8D4WevZhIJ2fwZ9e3IPx6ud3cU/edit#heading=h.gzfkq5p7slka>

## SBR Setup

[https://docs.google.com/document/d/1OHAbMBOcD4cqQUFgRFBmdCfeTeo6FHd\\_ihy7mKqqT-A/edit#heading=h.c8jx36i04ge9](https://docs.google.com/document/d/1OHAbMBOcD4cqQUFgRFBmdCfeTeo6FHd_ihy7mKqqT-A/edit#heading=h.c8jx36i04ge9)

Check if working: <https://demo.localtropy.com/messages>

From Admin portal enable features