

# Deploying Dockerized Rails App to AWS EC2/ECS #2

Ankit, Trung

Growth Session #19 - October 11-12 2018



# Deployment Setup

- Mina as deployment automation tool
- Puma as app server(Rails 5 default)

EC2 T2 small instance

vcliq-aws-ter... i-0e7166a0ef8b66677 t2.small ap-southeast-1a running 2/2 checks ... None ec2-54-169-53-23.ap-s... 54.169.53.23 - vcliq-deploy-key disabled

RDS

vcliqdb PostgreSQL available 1.33% 1 Connections none db.t2.micro vpc-6f9b120a No

In total 3 containers for Redis, App and Nginx.

```
ubuntu@ip-172-31-2-161:/var/www/vcliq.com/current$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
581383e06e81	6_nginx	"nginx -g 'daemon of..."	39 seconds ago	Up 38 seconds	0.0.0.0:80->80/tcp	6_nginx_1
1500dea2bf92	vcliq-web:development	"/bin/sh -c ./bin/st..."	About a minute ago	Up 38 seconds	3000/tcp	6_staging_1
25cd6699e6b2	redis:4.0.9	"docker-entrypoint.s..."	About a minute ago	Up About a minute	0.0.0.0:6379->6379/tcp	vcliq-web_redis

# Docker Compose File

```
redis:
  image: redis:4.0.9
  container_name: vcliq-web_redis
  ports:
    - "6379:6379"

staging:
  build:
    context: .
    cache_from:
      - ${DOCKER_IMAGE}:${BRANCH_TAG}
  image: ${DOCKER_IMAGE}:${BRANCH_TAG}
  stdin_open: true
  tty: true
  volumes:
    - /var/www/vcliq-web
  expose:
    - "3000"
  env_file: .env.staging

nginx:
  build:
    context: .
    dockerfile: Dockerfile-nginx
  links:
    - staging
  ports:
    - "80:80"

volumes:
  ruby-bundle:
```

## Why use nginx?

Even a configuration like this could work and Puma could directly speak to HTTP traffic

```
staging:
  build:
    context: .
    cache_from:
      - ${DOCKER_IMAGE}:${BRANCH_TAG}
  image: ${DOCKER_IMAGE}:${BRANCH_TAG}
  stdin_open: true
  tty: true
  volumes:
    - /var/www/vcliq-web
  ports:
    - "80:3000"
  env_file: .env.staging
```

# Why use nginx?

## Some examples:

- Static redirects- you could setup your nginx to redirect all `http` traffic to the same url with `https` . This way such trivial requests will never hit your app server.
- Multipart upload- Nginx is better suited to handle multipart uploads. Nginx will combine all the requests and send it as a single file to puma.
- Serving static assets- It is recommended to serve static assets (those in `/public/` endpoint in rails) via a webserver without loading your app server.
- There are some basic DDoS protections built-in in nginx.

# Mina Script

```
desc "Deploys the current version to the server."
task :deploy => :environment do
  deploy do
    invoke :'git:clone'

    to :launch do
      invoke :'docker:build'
      invoke :'docker:stop' # stop the previous container
      invoke :'docker:run' # run new container with released code
    end
  end
end

namespace :docker do
  set :image_name, ENV.fetch("IMAGE_NAME", "vcliq-web")
  set :container_name, ENV.fetch("CONTAINER_NAME", "vcliq-web:development")

  set :configuration_volume, "#{deploy_to}/#{shared_path}/config:/config"
  set :log_volume, "#{deploy_to}/#{shared_path}/log:/log"

  desc "Build docker image"
  task build: :environment do
    queue "cd #{deploy_to}/#{current_path}"
    queue "docker-compose -f docker-compose.yml build"
  end

  desc "Start docker container"
  task run: :environment do
    queue "docker-compose -f docker-compose.yml up"
  end

  desc "Stop docker container"
  task stop: :environment do
    queue "if [ ! -z \"$(docker ps | grep '#{container_name}')\" ]; then docker stop #{container_name}; docker rm -f #{container_name};"
  end

  desc "Remove all stop container"
  task clean_containers: :environment do
    queue "sudo docker rm $(docker ps -a -q)"
  end
end
```

## Progress and Demo

1. A lot of setup had to be done again as the previous instance was down(setting up instance and installing docker)
2. Only install docker on the instance and leave the remaining dependencies to the dockerfile.
3. Database connection to RDS is working.
4. Processes up and running and a deployed version of vcliq looks like:

```
ubuntu@ip-172-31-2-161:/var/www/vcliq.com/current$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
581383e06e81	6_nginx	"nginx -g 'daemon of..."	39 seconds ago	Up 38 seconds	0.0.0.0:80->80/tcp	6_nginx_1
1500dea2bf92	vcliq-web:development	"/bin/sh -c ./bin/st..."	About a minute ago	Up 38 seconds	3000/tcp	6_staging_1
25cd6699e6b2	redis:4.0.9	"docker-entrypoint.s..."	About a minute ago	Up About a minute	0.0.0.0:6379->6379/tcp	vcliq-web_redis

## Further Steps

1. Fix the asset precompilation issues.
2. Check other integrations such as sidekiq.
3. Complete mina deployment scripts.
4. Integrate Elastic IP with EC2 Instance
  - Extremely important as if the instance restarts, the IP of the instance would change.
5. Configure logs backup to S3 and application monitoring.
6. Look at other AWS services like ECS.



# Thanks!

Contact Nimbl3

[hello@nimbl3.com](mailto:hello@nimbl3.com)

399 Sukhumvit Road, Interchange 21  
Klongtoey nua, Wattana  
Bangkok 10110

28C Stanley St,  
Singapore 068737

20th Floor, Central Tower  
28 Queen's Road  
Central, Hong Kong

[nimbl3.com](http://nimbl3.com)

