

Oma rakendusele pideva kohaletoimetamise toe lisamine

Dockeri install

1. Installi docker (<https://docs.docker.com/engine/install/ubuntu/>)

```
sudo apt-get update

sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release

sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /etc/apt/keyrings/docker.gpg

echo \
    "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-
compose-plugin
```

2. Anna ubuntu kasutajale õigus Dockerit kasutada nagu juhendis (<https://docs.docker.com/engine/install/linux-postinstall/>)

```
sudo groupadd docker
sudo usermod -aG docker $USER
newgrp docker
```

3. Installi awscli

```
sudo apt update
sudo apt install awscli
```

4. Pane paika AWS ligipääsu võimalused

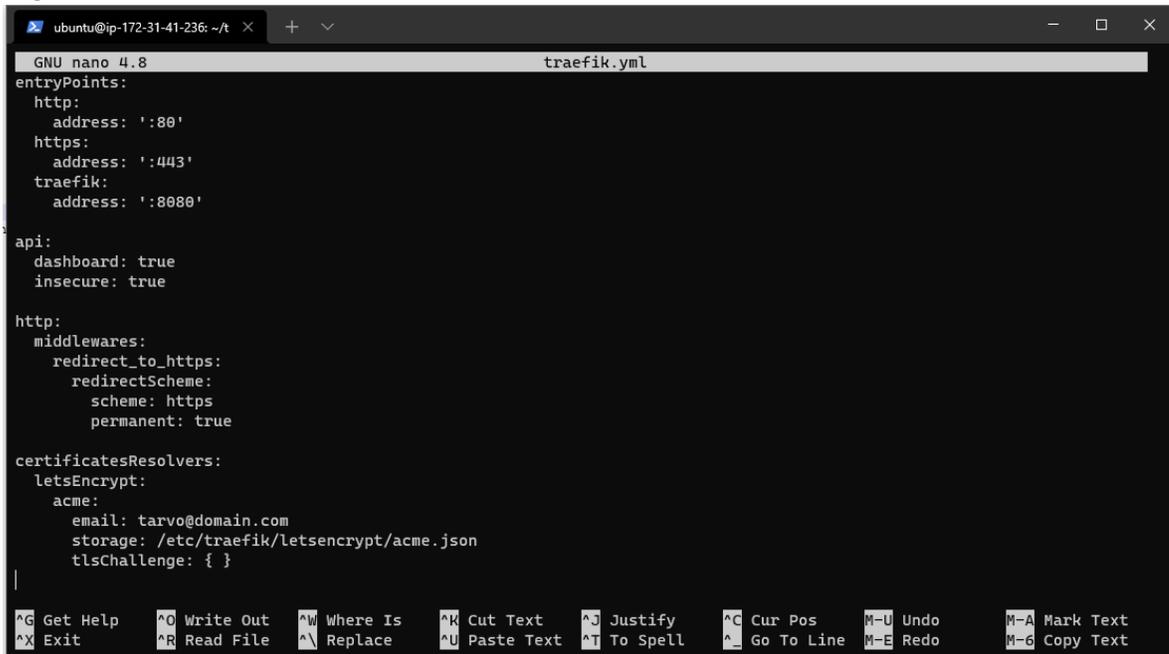
```
aws configure
```

5. Logi konteineri repositooriumisse sisse käsuga.

```
aws ecr get-login-password | sudo docker login --username AWS --password-stdin 662032419283.dkr.ecr.eu-north-1.amazonaws.com
```

Traefik

1. Loo AWS masinas kaust ~/traefik.
2. Lisa kausta traefik Moodle-i arhiivist traefik failid traefik.yml ja docker-compose.yml.
3. Muuda traefik.yml-s emaili. Valitud emailile tuleb teade kui HTTPS sertifikaadid hakkavad aeguma.



```
ubuntu@ip-172-31-41-236: ~/traefik
GNU nano 4.8 traefik.yml
entryPoints:
  http:
    address: ':80'
  https:
    address: ':443'
  traefik:
    address: ':8080'
api:
  dashboard: true
  insecure: true
http:
  middlewares:
    redirect_to_https:
      redirectScheme:
        scheme: https
        permanent: true
certificatesResolvers:
  letsEncrypt:
    acme:
      email: tarvo@domain.com
      storage: /etc/traefik/letsencrypt/acme.json
      tlsChallenge: { }
```

4. Loo Traefikule Dockeri võrk käsuga

```
docker network create traefik
```

5. Käivita docker käsuga

```
docker-compose up -d
```

Ettevalmistused serveris rakenduse käivitamiseks

1. Loo kaust ~/app.
2. Lae Moodle-i arhiivist docker failid .env ja docker-compose.yml app kausta
3. Muuda .env failis muutujaid vastavalt vajadusele. Minimaalselt on vaja määrata DOMAIN väärtuseks enda projekti domeen ja BACK_END_IMAGE ja FRONT_END_IMAGE väärtusteks enda AWS image-te adressid. API_URL_NAME peaks olema NUXT_PUBLIC_ ja lõpp peaks olema

sama kui muutuja nimi nuxt.config.ts failis runtimeConfig public all ainult suurte tähtedega ja alakriipsudega

Pidev kohaletoimetamine

Pideva kohaletoimetamise üles seadmiseks laseme GitLab-i runneril läbi ssh käima panna docker-compose up käsu

1. Loo oma arvutis runnerile ssh võti käsuga

```
ssh-keygen -t ed25519 runner
```

2. Lisa loodud public key AWS masinas faili ~/.ssh/authorized_keys.
3. Lisa loodud private key sisu GitLabis muutujasse SSH_PRIVATE_KEY.

Variable	Value	Type	Environment scope	Actions
AWS_SECRET_ACCESS_KEY	*****	Variable	All (default)	[Edit] [Delete]
SSH_PRIVATE_KEY	*****	Variable	All (default)	[Edit] [Delete]

4. Lisa oma projektis .gitlab-ci.yml faili deploy stage näidisprojekti näitel
5. Muuda .gitlab-ci.yml failis deploy -> script all ssh address enda AWS virtuaalmasina addressiks.
6. Kommiti ja pane üles.
7. Käivita Gitlabis deploy pipeline.
8. Vaata oma veebilehte aadressil MINU_PROJEKTI_NIMI.itb2203.tautar.ee.