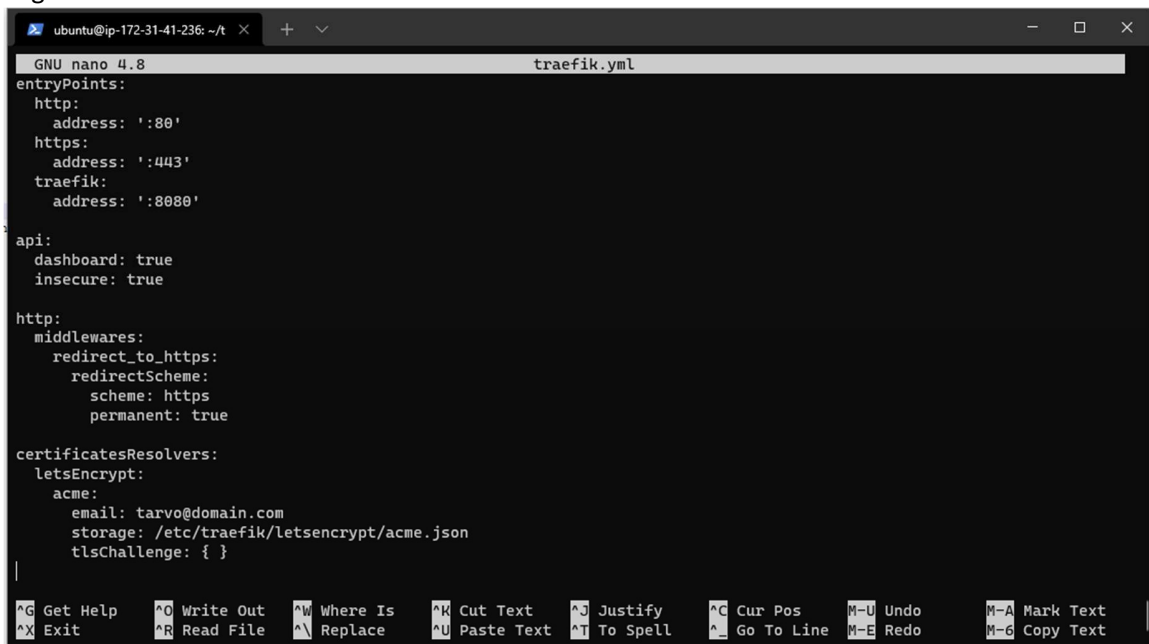


Oma rakendusele pideva kohaletoimetamise toe lisamine

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.



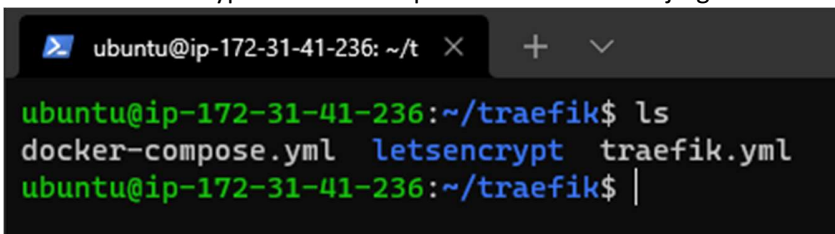
```
ubuntu@ip-172-31-41-236: ~/t
GNU nano 4.8 traefik.yml
entryPoints:
  http:
    address: ':80'
  https:
    address: ':443'
  traefik:
    address: ':8080'

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 kaust letsencrypt. Tulemusena peaksid olema olema järgmised failid.



```
ubuntu@ip-172-31-41-236: ~/t
ubuntu@ip-172-31-41-236: ~/traefik$ ls
docker-compose.yml  letsencrypt  traefik.yml
ubuntu@ip-172-31-41-236: ~/traefik$ |
```

5. Loo Docker võrk käsuga

```
docker network create traefik
```

6. Käivita docker käsuga

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

Käsitsi rakenduse üles seadmine

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. Installi awscli

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

3. 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
```

4. Kommiti ja pane muudatused üles ning kontrolli, et luuakse uus Docker-i image AWS-s.
5. Loo kaust ~/app.
6. Lae Moodle-i arhiivist docker failid .env ja docker-compose.yml app kausta
7. 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 aadressid.
8. Loo kaust ~/app/config/frontend.
9. Lisa kausta oma projekti FrontEnd/public/config kaustast runtime-config.json. ~/app kaustal peaks nüüd olema järgmine struktuur.
10. Määra runtime-config.json-is API asukohaks api.MINU_RAKENDUS.itb2203.tautar.ee.

```
ubuntu@ip-172-31-46-50:~/app$ find . -type f
./docker-compose.yml
./conf/frontend/runtime-config.json
./env
ubuntu@ip-172-31-46-50:~/app$
```

11. Mine tagasi kausta ~/app.
12. Käivita docker käsuga.

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

13. Vaata, kas rakendus toimib. Ava https://MINU_PROJEKT.itb2203.tautar.ee/.

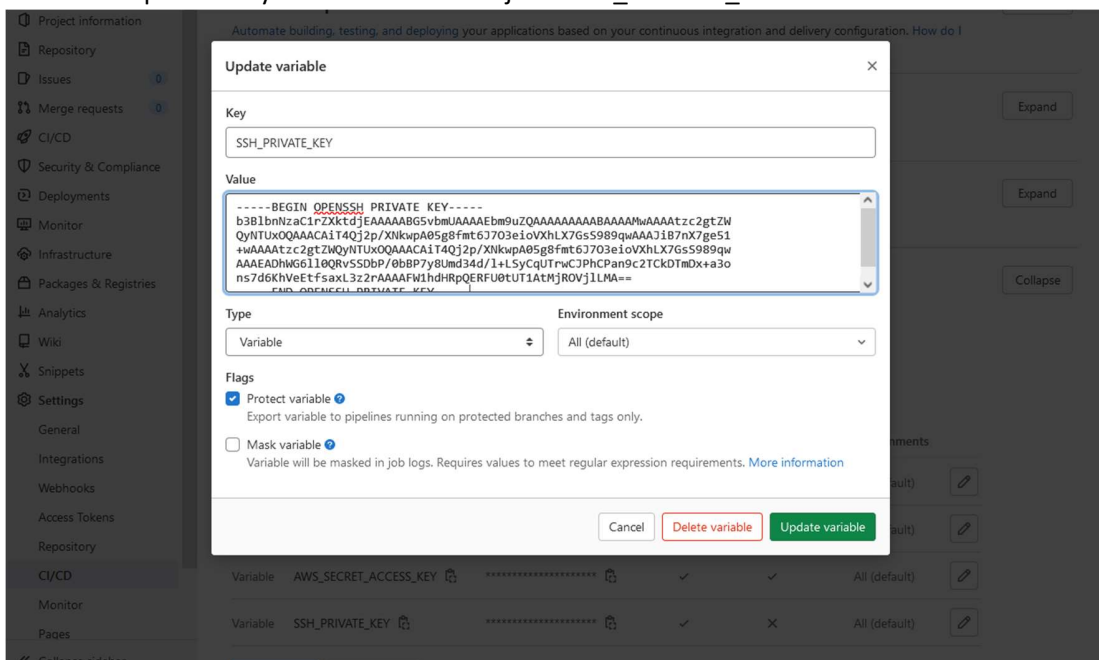
Pidev kohaletoimetamine

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`.



4. Lisa oma projektis .gitlab-ci.yml faili deploy stage näidisprojekti näitel
5. Muuda .gitlab-ci.yml failis deploy -> script all ssh aadress enda AWS virtuaalmasina aadressiks.
6. Kommiti ja pane üles mõni muudatus.
7. Kontrolli, et deploy pipeline toimib.

W

Workout CI CD

Project information

Repository

Issues0

Merge requests0

CI/CD

Pipelines

Editor

Jobs

Schedules

Security & Compliance

Deployments

Monitor

Infrastructure

Packages & Registries

Analytics

Wiki

Snippets

Mattias.Koosalu > Workout CI CD > Pipelines > #64969

passed Pipeline #64969 triggered 28 minutes ago by Mattias.Koosalu

Lisa CI/CD tugi

🕒 2 jobs for **master** in 3 minutes and 34 seconds (queued for 3 seconds)

🚩 latest

🔑 5f9b0222 🔗

🔗 No related merge requests found.

Pipeline Needs Jobs 2 Tests 0

Build

Deploy

✅ build



⚙️ deploy

