

ePrescription og Patient Summary driftsvejledning

Idriftsættelse

For at din server kan køre løsningen, skal følgende forudsætninger være opfyldt:

Adgang til Testa netværket

Serveren skal have adgang til Testa netværket. Følg vejledningen "Tilslutning til TESTA" i appendiks for at konfigurere denne adgang. (Bemærk: tidskrævende med ugers turnaround)

Installation af Docker

Docker skal være installeret på serveren. Følg vejledningen "Installation af Docker" i appendiks for at installere og konfigurere Docker korrekt.

Firewall konfiguration

Firewall-indstillingerne på serveren skal tillade trafik på følgende porte mod TESTA og hvorfra serveren administreres:

- Port 443
- Port 8080
- Port 1442
- Port 1443
- Port 2443
- Port 3443
- Port 6443
- Port 8092
- Port 8443

Sørg for, at disse porte er åbne i din firewall.

Yderligere overvejelser

Vi kan yderligere overveje om vi skal implementere følgende:

- **Sikkerhedsopdateringer:** Hold serveren opdateret med de nyeste sikkerhedsopdateringer.
- **Overvågning:** Implementer overvågning for at holde øje med serverens ydeevne.
- **Backup:** Sørg for, at der er en backup-løsning på plads.

Opsætningsguide

Følg disse trin for at opsætte systemet:

Klon Git Repository

Start med at klon følgende Git repository:

```
git clone https://github.com/Sundhedsdatastyrelsen/ehdsi
```

Konfigurer NCP Mappen

1. Gå ind i **NCP** mappen.
2. Kopier alt indhold fra **env_default** mappen til roden af **NCP** mappen.
3. Indsæt de korrekte produktionsværdier i de relevante filer.
4. Placer de relevante keystores i **keystores** mappen.

Log ind på Docker

Log ind på Docker gennem GitHub Container Registry:

```
docker login ghcr.io
```

Start Docker NCP

Start Docker NCP:

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

Konfigurer Country-A-Service

1. Skift til **country-a-service** mappen.
2. Kopier **.env.defaults** filen til **.env** og udfyld den med de nødvendige værdier.

Start Country-A-Service

Start Country-A-Service:

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

Vigtige Produktionsopsætninger

For at undgå at få for meget data ud i logs, skal følgende værdier være sat i **.env** filen for NCP:

```
automated.validation=false  
WRITE_TEST_AUDITS=false  
Server.ehealth.mode=PRODUCTION
```

Deploy af nye versioner

Når der skal lægges nye versioner af denne enhed, skal køres

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

Opsætningsguide - Frabedelsesservice

Klon Git Repository

Start med at klon følgende Git repository:

```
git clone https://github.com/Sundhedsdatastyrelsen/FSEU
```

Start Docker

Start Docker:

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

Deploy af nye versioner

Når der skal lægges nye versioner af denne enhed, skal køres

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

Drift

Logs

For at hente logs for en specifik service, brug navnet på servicen fra `docker-compose.yml` filen. Kør følgende kommando i samme sti som `docker-compose.yml`:

```
docker compose logs [containernavn] -f
```

Flaget `-f` er til at følge loggene i realtid. Udelad flaget, hvis du kun vil se de seneste logs.

Debugging

Debugging af en Enkelt Container

For at debugge en enkelt container, tilføj følgende værdi under **environment** i **docker-compose.yml**:

```
-agentlib:jdwp=transport=dt_socket,address=*:5005,server=y,suspend=n
```

Åbn den valgte port under **ports** i **docker-compose.yml** og i firewallen. Du kan derefter forbinde til containeren med Remote JVM Debug i IntelliJ.

Container Debugging

For at debugge direkte inde i containeren, start en shell med:

```
docker exec -it [containernavn] /bin/bash
```

Erstat **/bin/bash** med **/bin/sh**, hvis bash ikke er installeret. Find containernavnet i **docker-compose.yml** eller ved at køre **docker ps**.

Herefter kan du køre kommandoer i containerens kontekst

Appendiks

Installation af Docker

Installationen af Docker foregår relativt simpelt på Linux servere. Verificer gerne, at alle URL'er i denne vejledning er korrekte inden start, i tilfælde af gamle domæner osv.

Forberedelse af serveren

Først gøres serveren klar med eventuelle dependencies til at hente Docker:

```
sudo apt update && sudo apt upgrade  
sudo apt install -y ca-certificates curl gnupg lsb-release
```

Tilføjelse af Dockers GPG-nøgle

Derefter tilføjes Dockers GPG-nøgle, så vi stoler på deres server:

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

Tilføjelse af Docker-repository

Tilføj Docker-repository til serveren:

```
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
```

Installation af Docker

Installer Docker:

```
sudo apt update  
sudo apt install -y docker-ce docker-ce-cli containerd.io
```

Start af Docker

Start Docker:

```
sudo systemctl start docker  
sudo systemctl enable docker
```

Kør Docker som en ikke-root bruger

For at køre Docker som en ikke-root bruger, skal du tilføje din bruger til Docker-gruppen:

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

Verificering af installationen

For at verificere, at Docker er installeret korrekt, kan du køre følgende kommando:

```
docker run hello-world
```

Denne kommando vil downloade et test-image og køre det i en container. Hvis alt er installeret korrekt, vil du se en velkomstbesked fra Docker.

Konfiguration af Docker til at starte ved boot

For at sikre, at Docker starter automatisk ved boot, kør:

```
sudo systemctl enable docker
```

Sikkerhedsopdateringer

Det er vigtigt at holde Docker opdateret for at sikre, at du har de nyeste sikkerhedsopdateringer. Du kan opdatere Docker ved at køre:

```
sudo apt update  
sudo apt upgrade
```

Fejlfinding

Hvis du støder på problemer under installationen, kan du tjekke Docker-loggene for fejlmeddelelser:

```
sudo journalctl -u docker
```

Du kan også søge efter løsninger på almindelige problemer på Docker's officielle dokumentation eller på fora som Stack Overflow.

Tilslutning til TESTA

For at en server tilsluttes TESTA, skal der indsendes en blanket til godkendelse hos DXC. Hostes serveren hos SIT, skal det klares via dem, og det kan ikke klares via en formular - men skal bruge en dialog, hvor de internt udfylder en formular.

Forbindelsen i dag er sat op med direkte port forwarding fra routeren til vores server - hvilket gør det risikabelt at tage serveren ned, og svært at lave om på. Vi ønsker os en løsning med en load balancer i mellem, men er opmærksomme på at vi måske selv må stå for det.