

Operato ECS 설치 가이드

설치 사양

- OS: 도커 설치 가능한 OS로서 본 문서에서는 Ubuntu 기반으로 설명이 되어 있습니다.



Docker가 설치 가능한 Linux 기반 환경에서는 대부분 설치가 가능합니다. 본 매뉴얼에서는 Ubuntu 22에서 검증된 내용들을 명시합니다. 도커 설치가 가능한 다른 Linux OS에서도 설치가 가능하고 OS 종류와 사용자 권한에 따라서 명령어의 변경이 필요할 수 있습니다.



CPU와 메모리 사양은 저사양 PC에서도 동작 가능하지만, 제작하고자 하는 보드 구성 등을 고려해서 적절한 사양을 선택합니다.

도커 엔진 설치

Install Docker Engine on Ubuntu



Jumpstart your client-side server applications with Docker Engine on Ubuntu. This guide details prerequisites and multiple methods to install.

 <https://docs.docker.com/engine/install/ubuntu/>

도커 엔진 설치 전 셋업

아래 명령어를 순차적으로 입력합니다.

```
# apt package upage
sudo apt-get update

sudo apt-get install ca-certificates curl gnupg
```

```
# add Docker's official GPG key
sudo install -m 0755 -d /etc/apt/keyrings

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

sudo chmod a+r /etc/apt/keyrings/docker.gpg

# setup the repository
echo \
    "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
    "$(. /etc/os-release && echo "$VERSION_CODENAME)" stable" | \
    sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

도커 엔진 설치

```
sudo apt-get update

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

설치 확인

docker --help 명령어를 실행합니다.

```
docker --help
```

아래와 같이 도움말 표시되면 설치가 완료된 것입니다.

```
Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers
```

Common Commands:

run	Create and run a new container from an image
exec	Execute a command in a running container
ps	List containers
...	

docker-compose 설치

여러 도커 이미지를 동시에 실행 및 관리하기 위한 docker-compose를 설치합니다.

설치 명령어 실행

```
sudo curl -L https://github.com/docker/compose/releases/download/v2.20.2/docker-compose-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose
```

권한 변경

```
sudo chmod +x /usr/local/bin/docker-compose
```

설치 확인

```
docker-compose --version
```

아래와 같이 버전 표시되면 설치가 완료된 것입니다.

```
Docker Compose version v2.20.2
```

sudo 없이 도커 실행되도록 변경

기본적으로 어플리케이션 설치 및 실행에 필요한 스크립트들이 sudo 없는 docker 명령어로 구성되어 있기 때문에 설치 이전에 반드시 아래 명령어를 수행합니다. 명령어 수행 이후에는 시스템을 재부팅해야 합니다.

```
sudo usermod -aG docker ${USER}
```



해당 명령어 실행 후, 반드시 시스템 재부팅이 되어야 적용이 됩니다. 만약 OS 혹은 서버의 설치 환경에 의해서 sudo 관련 설정 변경이 어려울 경우에는 차후 설치가 될 어플리케이션 스크립트들 내의 docker와 docker-compose에 대한 명령어에 대해서 sudo를 추가해서 사용해도 됩니다.

Operato ECS 설치



아래 설치 명령어를 실행하기 이전에 설치하고 하는 디렉토리를 생성해서 해당 디렉토리로 먼저 이동합니다. 모든 설치 파일들은 해당 디렉토리에 설치됩니다.

설치 명령어 실행

```
curl -fsSL https://raw.githubusercontent.com/things-factory/things-factory/master/packages/operato-ecs/installer/install.sh | bash -s
```

위 명령어를 수행하면, 도커 운용에 필요한 여러 스크립트들을 다운로드 받고, 이후에 필요한 도커 이미지들을 아래와 같이 다운로드

```
> curl -fsSL https://raw.githubusercontent.com/things-factory/things-factory/master/packages/operato-ecs/installer/install.sh | bash -s
config.production.js create
  % Total    % Received % Xferd  Average Speed   Time    Time
me      Time      Current
                                 Dload  Upload  Total  Sp
ent      Left  Speed
100 1579 100 1579    0    0  3296      0 --:--:-- --:--
-:-- --:--:-- 3303
start.sh create
  % Total    % Received % Xferd  Average Speed   Time    Time
me      Time      Current
```

```

ent      Left  Speed
100    260 100    260    0    0    661      0 --:--:-- --:--
-:-- -:--:--    659
stop.sh create
  % Total      % Received % Xferd  Average Speed   Time    Ti
me      Time  Current
                                Dload  Upload  Total  Sp
ent      Left  Speed
100     19 100     19    0    0     40      0 --:--:-- --:--
-:-- -:--:--     40
upgrade.sh create
  % Total      % Received % Xferd  Average Speed   Time    Ti
me      Time  Current
                                Dload  Upload  Total  Sp
ent      Left  Speed
100    134 100    134    0    0    286      0 --:--:-- --:--
-:-- -:~:~:~    286
migrate.sh create
  % Total      % Received % Xferd  Average Speed   Time    Ti
me      Time  Current
                                Dload  Upload  Total  Sp
ent      Left  Speed
100     66 100     66    0    0    159      0 --:~:~:~ --:~:~
-:~:~ -:~:~:~    160
docker-compose.yml create
  % Total      % Received % Xferd  Average Speed   Time    Ti
me      Time  Current
                                Dload  Upload  Total  Sp
ent      Left  Speed
100   1038 100   1038    0    0   2596      0 --:~:~:~ --:~:~
-:~:~ -:~:~:~    2601
scheduler-config.yaml create
  % Total      % Received % Xferd  Average Speed   Time    Ti
me      Time  Current
                                Dload  Upload  Total  Sp
ent      Left  Speed
100     14 100     14    0    0     28      0 --:~:~:~ --:~:~

```

```

-:-- -:--:--      28
latest: Pulling from hatiolab/operato-ecs
fea1432adf09: Already exists
5651b5803b18: Already exists
3873416e6a33: Already exists
8a142b8b0e69: Already exists
3f02c49845cb: Already exists
844f6c465911: Already exists
ba5ea3d48002: Already exists
929ce2e05565: Already exists
fac7f86153bd: Already exists
78750bc2d090: Already exists
1b12ca95915c: Already exists
5da44e050698: Already exists
f32049955510: Already exists
47612dbdaaf5: Already exists
42ac1e5e20c0: Already exists
32a237c7a749: Already exists
d9bb0b453e46: Already exists
db3dae423e27: Already exists
f7864ea845ca: Already exists
1f6c7172fc4d: Already exists
83e6ce415524: Already exists
803650403507: Already exists
d0f2160df891: Already exists
4f4fb700ef54: Already exists
20a2a8a33ec2: Already exists
83ae54f18e7a: Already exists
0df11d3e9ac3: Already exists
2d4765dfdbd1: Pull complete
b1b4341e3809: Pull complete
eb070add1251: Pull complete
Digest: sha256:04046f4b1e9980fba69f13833e96510f1d32001ce050
22ed30a8f9af5ef7e8af
Status: Downloaded newer image for hatiolab/operato-ecs:lat
est
docker.io/hatiolab/operato-ecs:latest
...

```

설치 완료 후 확인 사항

설치 완료 후 다음과 같은 디렉토리로 구성되어 있는 지 확인한다.

```
> ls -al                                     jinwoncho
i@JinWonui-MacBookPro
total 64
drwxr-xr-x@ 10 jinwonchoi  staff   320  7 18 16:23 .
drwxr-xr-x  27 jinwonchoi  staff   864  7 18 16:23 ..
-rw-r--r--@  1 jinwonchoi  staff    14  7 18 16:23 .env
-rw-r--r--@  1 jinwonchoi  staff  1579  7 18 16:23 config.p
roduction.js
-rw-r--r--@  1 jinwonchoi  staff  1038  7 18 16:23 docker-c
ompose.yml
-rwxr--r--@  1 jinwonchoi  staff    66  7 18 16:23 migrate.
sh
-rw-r--r--@  1 jinwonchoi  staff    14  7 18 16:23 schedule
r-config.yaml
-rwxr--r--@  1 jinwonchoi  staff   260  7 18 16:23 start.sh
-rwxr--r--@  1 jinwonchoi  staff    19  7 18 16:23 stop.sh
-rwxr--r--@  1 jinwonchoi  staff   134  7 18 16:23 upgrade.
sh
```

라이선스 파일 준비

제공받은 라이선스 파일을 설치된 디렉토리에 복사합니다. 설치 시에 포함되어 있는 license.json 파일과 동일한 이름으로 변경해서 기존 파일을 대체합니다.

데이터베이스 마이그레이션

초기 어플리케이션 운용에 필요한 데이터를 준비하기 위해서 데이터 마이그레이션 작업이 필요합니다. 다음과 같이 순차적으로 명령어를 수행합니다.

```
# 어플리케이션 시작
./start.sh

# 마이그레이션 명령 수행
./migrate.sh

# 마이그레이션 종료 후, 어플리케이션 종료
./stop.sh
```

어플리케이션 실행

마이그레이션 작업이 완료되면, 다음과 같은 명령어로 어플리케이션을 실행합니다.

```
./start.sh
```

다음과 같이 수행되면 정상적으로 실행된 것입니다.

```
Warning: default port 3000
HOST_PORT : 3000
[+] Running 5/5
  ✓ Container db-operato-ecs           Started
  ✓ Container scheduler                Started
  ✓ Container operato-ecs-install-test-mosquitto-1 Started
  ✓ Container operato-ecs              Started
  ✓ Container operato-ecs-install-test-nginx-1   Started
```



기본 포트는 **4000**번으로 설정되어 있습니다.

만약 다른 포트로 서비스를 원한다면, start.sh 다음에 포트 번호를 명시합니다. 만약, 80포트로 서비스를 원하면 다음과 같이 실행합니다.


```
./start.sh 80
```

어플리케이션 업그레이드 및 버전 변경

스크립트를 활용한 최신 버전으로 변경

설치된 스크립트 중에 다음 스크립트를 통해서 최신 버전으로 업그레이드를 수행합니다.

```
./upgrade.sh
```

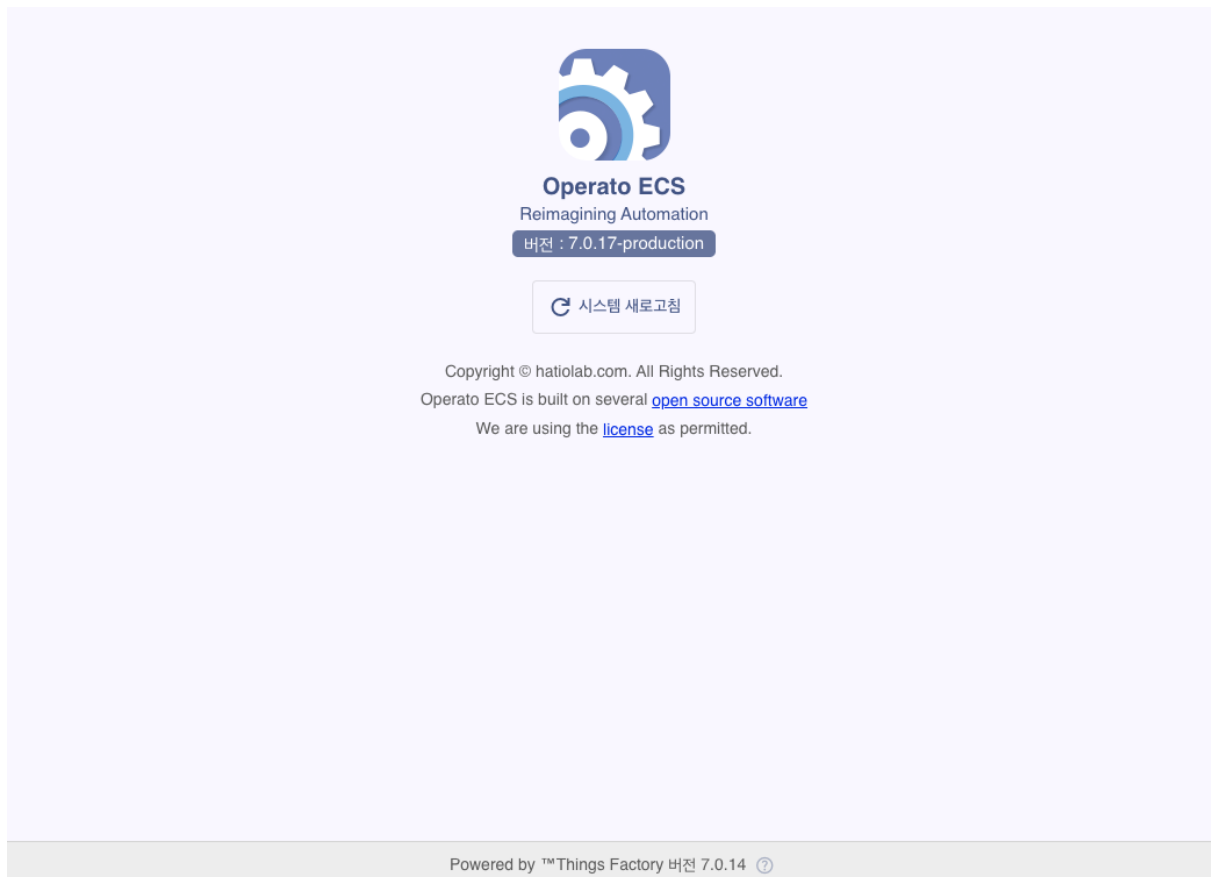
특정 버전으로 실행하고자 할 경우,

설치된 파일 중에 docker-compose.yml 파일에서 app의 image의 tag 부분을 특정 버전으로 수정합니다. 아래는 기존 latest로 명시된 부분을 0.0.1 으로 변경할 경우의 예입니다.


```
version: '3'
services:
  nginx:
    image: hatiolab/operato-nginx:latest
    ports:
      - ${HostPort}:80
    depends_on:
      - app
  app:
    build: .
    container_name: operato-ecs
    image: hatiolab/operato-ecs:latest
    privileged: true
    volumes:
      - ./logs:/app/logs
      - ./attachments:/app/attachments
      - ./config.production.js:/app/config.production.js
      - ./license.json:/app/license.json
    ports:
      - 4000:3000
```

```
depends_on:
  - postgres
  - mosquitto
  - scheduler
restart: always
```

업그레이드 혹은 docker-compose.yml 수정 후, **./start.sh**를 통해 어플리케이션을 시작하면, 변경된 버전으로 어플리케이션이 시작됨을 확인할 수 있습니다. 변경된 어플리케이션 버전은 시스템 정보 메뉴에서 확인이 가능합니다.



버전 변경 후 처음 브라우저로 실행을 하면 Progress WebApp의 특성으로 인해 수 초의 시간 동안 필요한 파일들을 변경한 이후에 아래 그림과 같이 **어플리케이션 재시작**을 요청합니다. 재시작을 누르면 새로운 버전으로 업데이트된 것을 확인하실 수 있습니다.

 어플리케이션이 업데이트되었습니다.

재시작

어플리케이션 종료

실행 중인 어플리케이션을 종료하려면 다음과 같이 수행합니다.

```
./stop.sh
```

다음과 같이 수행되면 정상적으로 종료된 것입니다.

```
[+] Stopping 5/5
✓ Container operato-ecs-install-test-nginx-1      Stopped
✓ Container operato-ecs                            Stopped
✓ Container scheduler                              Stopped
✓ Container db-operato-ecs                         Stopped
✓ Container operato-ecs-install-test-mosquitto-1  Stopped
```

브라우저를 통한 실행 확인

기본 포트(4000)로 실행했을 경우, 브라우저 주소 창에 다음과 같이 입력하면 실행을 확인할 수 있습니다.

http://{server ip 또는 domain}:4000

정상적으로 실행되면 다음과 같은 로그인 화면을 확인할 수 있습니다.



기본적으로 설정되어 있는 계정 정보는 다음과 같습니다.

이메일: admin@hatiolab.com

비밀번호: admin



첫 로그인에서 위 명시된 계정으로 로그인이 수행되지 않고 사용자가 없다는 여러 메시지가 확인되면 migration을 수행하지 않았기 때문입니다. 서버 실행을 중지(./stop.sh) 하고 마이그레이션을 수행합니다.

사용자 역할 생성 및 권한 할당

어플리케이션 초기 실행 시에 Admin 계정(admin@hatiolab.com)에 대해서 역할 생성 및 권한 할당 작업이 필수입니다. 해당 작업을 통해서 Admin 계정의 모든 어플리케이션 기능 수행이 가능해 집니다.

관리자 역할 생성

1. 우측 상단의 메뉴를 선택하고 '역할 관리'를 선택하면 다음과 같은 화면이 표시됩니다.

The screenshot shows the '역할 관리' (Role Management) page. At the top, there is a header bar with a home icon, the title '역할 관리', and a notification icon. Below the header, there is a form to create a new role. The form has two input fields: '역할 이름' (Role Name) and '역할 설명' (Role Description). To the right of these fields is a '생성' (Create) button. Below the form, there is a section titled '등록된 역할 리스트' (Registered Role List) which is currently empty.

2. 관리자 역할을 생성하기 위해서 다음과 같이 역할을 지정하고 생성합니다. 본 설명에는 역할 이름을 'ADMIN'으로 지정하였습니다.

The screenshot shows the '역할 관리' (Role Management) page after creating a new role. The '등록된 역할 리스트' (Registered Role List) section now contains one entry: 'ADMIN' with the description '관리자 계정 역할' (Administrator Account Role). The form above the list is still visible and empty.

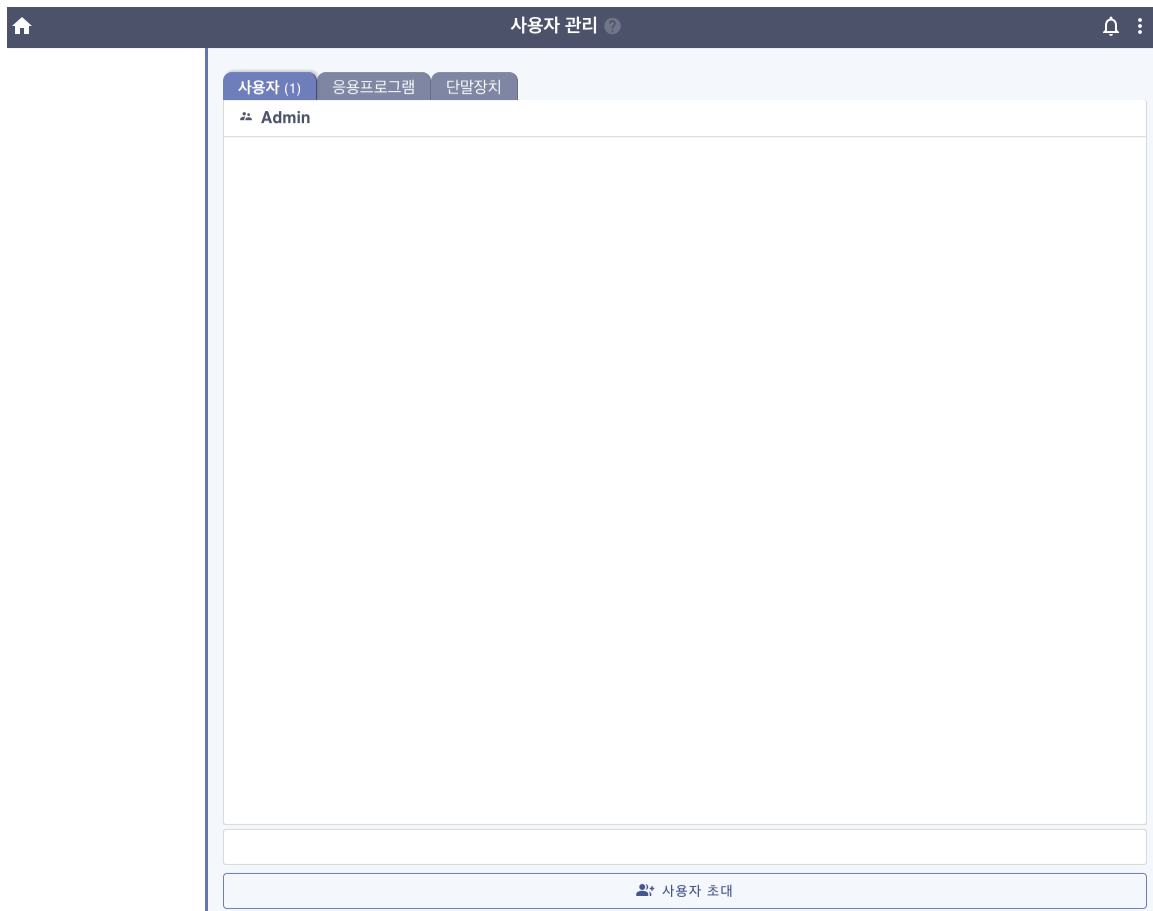
3. 생성된 역할(ADMIN)을 선택하면 다음과 같이 역할에 대한 권한을 설정할 수 있습니다.

4. ADMIN 역할에 대해서 'Check all'을 선택하고 저장을 누르면 관리자 계정에 대한 역할 생성이 완료됩니다.

Operato ECS 설치 가이드

관리자 역할 할당

1. 메뉴에서 '사용자 관리'를 선택하면 다음과 같은 화면이 표시됩니다.



2. Admin 계정(admin@hatiolab.com)을 선택하면 역할을 선택할 수 있고, 여기에서 앞에서 우리가 생성한 ADMIN 역할을 선택하고 저장을 누르면 Admin 계정을 통해서 다양한 기능을 활용할 수 있습니다.

다른 고려 사항들

- https 연결을 웹서버 설정이나 무중단을 위한 이중화 등의 작업들은 고객 환경 구성에 따라 작업이 필요할 수 있습니다. 기본 구성은 설치 시에 자체 설정된 nginx가 도커 기반으로 설치 되어 운용되고 있습니다. 자세한 사항은 `docker-compose.yml` 파일을 참고합니다.
- 데이터베이스는 기본적으로 어플리케이션 설치 시 자동으로 설치되는 도커 기반의 postgres 데이터베이스가 설치됩니다. 만약 별도의 postgres 데이터베이스를 설치해서 운용하고자 한다면, 설치된 파일 중에 `config.production.js` 파일의 데이터베이스 연결 파라미터를 수정합니다.

```
...
ormconfig: {
  name: "default",
  type: "postgres",
```



```
    host: "postgres",  
    port: 5432,  
    database: "postgres",  
    username: "postgres",  
    password: "abcd1234",  
    synchronize: true,  
    logging: false,  
  },  
  ...
```