



docker

4장 보충자료

Cloud를 Docker Registry로
사용하기

1-Private Registry의 필요성

- Custom Docker Image를 Registry에 저장해야 할 이유
 - Docker Host에 생성된 Custom Docker Image를 다른 docker host에서 사용하기 위해서 Docker Registry에 저장하는 것이다
 - Docker Registry는 Docker Hub와 Azure Container Registry 등이 사용된다
 - Docker Registry를 이용하려면 사용자 계정이 있어야 한다
 - Docker Hub에는 Public과 Private가 있고, Azure Container Registry는 Private만 있다
 - Private Registry에 저장된 docker image를 다운로드하려면 반드시 먼저 Login을 해야 한다

2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Container Registries에서 Registry 추가하기
 - All Services - "container"검색하여 container registries 선택하기
 - docker image를 저장할 서버인 Docker Registry를 생성한다

Home > Container registries > Create container registry

Container registries
Default Directory

+ Add Edit columns More

Help us improve Azure Container Registry.

Filter by name...

NAME ↑↓

Create container registry

* Registry name
songofsongs .azurecr.io ✓

* Subscription
Azure Pass

* Resource group
☐ Create new ☒ Use existing
SKRG

* Location
East US

* Admin user ⓘ
Enable Disable

* SKU ⓘ
Standard

2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Container Registries에서 Registry 추가하기
 - 생성한 songofsongs.azurecr.io라는 Container Registry에 접근할 서버 이름과 Key를 복사해 둔다

The screenshot shows the 'Access keys' page for a Container Registry named 'songofsongs'. The page is divided into a left sidebar and a main content area. The sidebar contains a search bar and a list of navigation items: Overview, Activity log, Access control (IAM), Tags, Quick start, Events, Settings, Access keys (highlighted with a red box and a red circle with the number 1), Locks, Automation script, Services, Repositories, and Webhooks. The main content area displays the 'Access keys' for the 'songofsongs' registry. It includes fields for 'Registry name' (songofsongs), 'Login server' (songofsongs.azurecr.io, highlighted with a red box and a red circle with the number 2), 'Admin user' (with 'Enable' and 'Disable' buttons), 'Username' (songofsongs), and a table of access keys. The table has two columns: 'NAME' and 'PASSWORD'. The first row is highlighted with a red box and a red circle with the number 3, showing 'password' and 'TRQw7tVxnnBOg0DPFUIscWW7+ygz09NX'. The second row shows 'password2' and 'xLGOUGb0jTjruWV72TjDNzsABpM9/aWB'. A green checkmark is placed next to the first row's password field.

Home > Container registries > songofsongs - Access keys

songofsongs - Access keys
Container registry

Search (Ctrl+/)

Overview
Activity log
Access control (IAM)
Tags
Quick start
Events
Settings
Access keys
Locks
Automation script
Services
Repositories
Webhooks

Registry name
songofsongs

Login server
songofsongs.azurecr.io

Admin user
Enable Disable

Username
songofsongs

NAME	PASSWORD
password	TRQw7tVxnnBOg0DPFUIscWW7+ygz09NX
password2	xLGOUGb0jTjruWV72TjDNzsABpM9/aWB

2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Custom Docker Image를 Azure Cloud로 Push하기
 - 앞에서 이미 Private Docker container registry 생성하였다
 - Docker container registry 확인하기
 - **All Services** -> **Container Registries** 검색 -> 생성된 것 확인
 - registry 주소와 Access key를 다시 확인하기

2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Custom Docker Image를 Azure Cloud로 Push하기
 - Azure Container Registry에 로그인하기
 - **docker login songofsongs.azurecr.io -u songofsongs -p qp9huVty+9X9cY**
 - Docker 공식 이미지인 Nginx를 다운로드하기
 - **docker pull nginx**
 - **docker images nginx**
 - Nginx container 시작하기
 - **docker run -d --name nginxtest --rm -p 8888:80 nginx**
 - **--rm:** 혹시 nginxtest라는 container를 실행 후 사용하다가 exit를 하면 자동으로 container를 삭제하는 옵션이다
 - **docker ps**

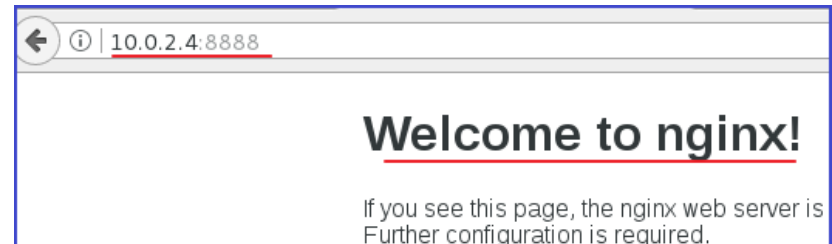
2-Azure에 Container Registry 생성하기

- Container Registries 생성하기

- Custom Docker Image를 Azure Cloud로 Push하기

- 두번째 Linux docker host에서 접속하기

- **http://10.0.2.4:8888**



- 첫번째 Linux docker host에서 Container 중지하기

- **docker stop nginxtest**

- **docker ps -a | grep nginxtest** (##없어짐. --rm으로 만들었기 때문)

- 첫번째 Linux docker host에서 공식 이미지의 Alias 생성하기

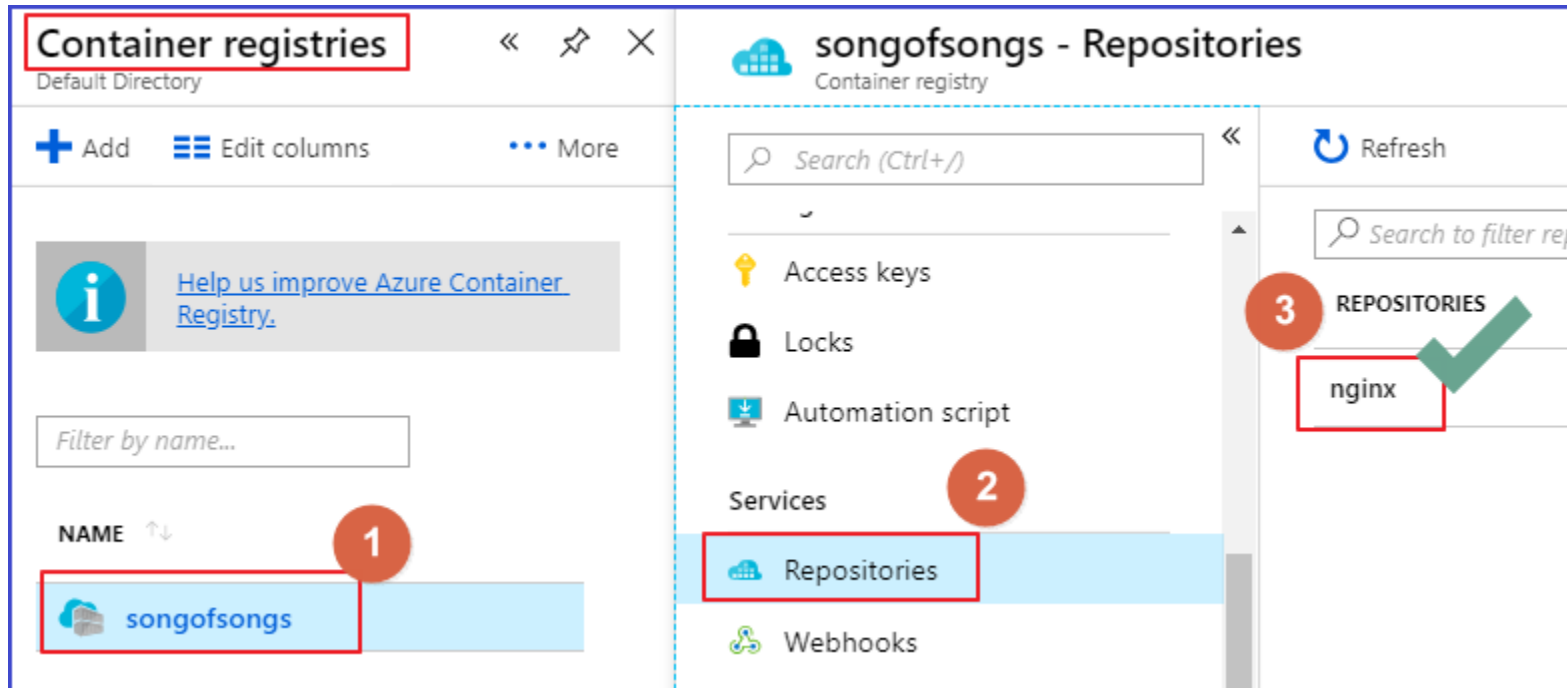
- **docker tag** nginx **songofsongs.azurecr.io/nginx**

- 여기서 songofsongs.azurecr.io/nginx에 주의를 한다

- **docker images**

2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Custom Docker Image를 Azure Cloud로 Push하기
 - `jesuswithme.azurecr.io/nginx` 이미지를 Azure Container Registry에 업로드하기
 - **docker push songofsongs.azurecr.io/nginx**



2-Azure에 Container Registry 생성하기

- Container Registries 생성하기
 - Custom Docker Image를 Azure Cloud로 Push하기
 - 두번째 Linux docker host에서 Azure Container Registry에 저장된 nginx 이미지를 다운로드하여 실행하기
 - **ssh 10.0.2.5**
 - **docker images** (##nginx 이미지가 없다)
 - **docker login songofsongs.azurecr.io -u songofsongs -p qp9huVty+9X9cY**
 - **docker pull songofsongs.azurecr.io/nginx**
 - **docker images | grep jesuswithme**
 - **docker run -d --rm -p 8888:80 songofsongs.azurecr.io/nginx**
 - 첫번째 Linux에서 접속해 보기
 - **http://10.0.2.5:8888**
 - **Ctrl + x**
 - **docker ps**

