# 

# Dockerizing UAA and K8

Author Greg Portnov 29.08.2016 v1

Contents

[Dockerizing UAA and K8 1](#_Toc460242413)

[Docker Build 2](#_Toc460242414)

[Kubernets 4](#_Toc460242415)

# Docker Build

1. Download UAA products to your local FS path

e.g /home/admin/

1. UAA structure and sources to build docker image:

├── Dockerfile

├── jdk1.8.0\_91

├── tomcat

├── apache-tomcat-8.0.35

├── cloudfoundry-identity-api-3.4.0.war.zip

   ├── cloudfoundry-identity-app-3.4.0.war.zip

│   ├── cloudfoundry-identity-uaa-3.4.0.war.zip

│   ├── postgresql-9.4.1209.jre6.jar

│   ├── server.xml

│   └── setenv.sh

├── uaa

│   ├── config

│   │   ├── log4j.properties

│   │   └── login.yml

│   ├── log

│   └── uaa.yml

1. Tomcat changes

Prerequisites: download apache-tomcat-8.0.35

* 1. setenv.sh copy from git docker/uaa-nurego/tomcat/
  2. startup.sh

export CATALINA\_OPTS="$CATALINA\_OPTS $(jolokia\_opts) -Djava.security.egd=file:/dev/./urandom"

/opt/tomcat/bin/catalina.sh run

* 1. catalina.sh, daemon.sh, tool-wrapper.sh

change setenv 🡪/mnt/setenv.sh

or copy from git tomcat/apache-tomcat-8.0.35/bin

* 1. Insert to server.xml or copy from git docker/uaa-nurego/tomcat/server.xml

<Resource name="jdbc/postgres" auth="Container"

type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"

url="jdbc:postgresql://${PGHOST}:${PGPORT}/${UAADB}"

username="${UAAUSER}" password="${UAAPASS}" maxTotal="20" maxIdle="10" maxWaitMillis="-1"/>

Replace “uaa-dev-new.nurego.com” with relevant value

1. Docker file

FROM 781127743430.dkr.ecr.us-east-1.amazonaws.com/nurego/ubuntu-16

COPY jdk1.8.0\_91 /opt/jdk1.8.0\_91

COPY tomcat/apache-tomcat-8.0.35 /opt/tomcat

COPY tomcat/server.xml /opt/tomcat/conf

COPY tomcat/cloudfoundry-identity-api-3.4.0.war.zip /opt/tomcat/webapps/api.war

COPY tomcat/cloudfoundry-identity-app-3.4.0.war.zip /opt/tomcat/webapps/app.war

COPY tomcat/cloudfoundry-identity-uaa-3.4.0.war.zip /opt/tomcat/webapps/ROOT.war

COPY tomcat/postgresql-9.4.1209.jre6.jar /opt/tomcat/lib

COPY uaa /opt/uaa

COPY tomcat/setenv.sh /mnt

CMD ["/bin/sh", "-c", "/opt/tomcat/bin/startup.sh"]

EXPOSE 8080

1. Docker image creation

* docker build -t uaa\_nurego -f Dockerfile .
* docker tag uaa\_nurego registry.nurego.com/nurego/uaa\_nurego
* docker push registry.nurego.com/nurego/uaa\_nurego

1. Docker compose - Uaa\_Dock.tgz download from

docker-compose.yml

version: '2'

services:

app:

image: "781127743430.dkr.ecr.us-east-1.amazonaws.com/nurego/uaa\_nurego"

volumes:

- /home/admin/Test/Uaa\_Dock/config:/opt/uaa/config

- /home/admin/Test/Uaa\_Dock/tomcat/setenv.sh:/mnt/setenv.sh

# Kubernets

1. Use UAA products from local FS.

e.g put them to /home/admin/

1. Config Map

kubectl create configmap uaa.conf --from-file=/home/admin/uaa/uaa/uaa.yml --from-file=/home/admin/uaa/uaa/config/log4j.properties --from-file=/home/admin/uaa/uaa/config/login.yml

kubectl create configmap uaa.set --from-file=/home/admin/uaa/tomcat/setenv.sh

1. Yaml file uaa-pod.yml

apiVersion: v1

kind: Pod

metadata:

name: uaa

spec:

containers:

- name: uaa

image: registry.nurego.com/nurego/uaa\_nurego

#command: [ "/bin/sh", "-c", "mkdir -p /home/comfigmap/uaa" ]

#resources:

# requests:

# memory: "512Mi"

# cpu: "250m"

#limits:

# memory: "848Mi"

# cpu: "500m"

ports:

- containerPort: 8080

volumeMounts:

- name: config-volume

mountPath: /opt/uaa/config

- name: config-set

mountPath: /mnt

volumes:

- name: config-volume

configMap:

name: uaa.conf

- name: config-set

configMap:

name: uaa.set

restartPolicy: Never

imagePullSecrets:

- name: nurego-registry-secret

1. Deployment UAA

root@ip-172-20-0-9:/home/admin/Test/K8/Deployment# uaa-pod.yml.deployment

apiVersion: v1

kind: Service

metadata:

name: uaa

labels:

app: uaa

spec:

ports:

- port: 8080

selector:

app: uaa

tier: frontend

type: LoadBalancer

---

apiVersion: extensions/v1beta1

kind: Deployment

metadata:

name: uaa

labels:

app: uaa

spec:

replicas: 3

template:

metadata:

labels:

app: uaa

spec:

containers:

- image: 781127743430.dkr.ecr.us-east-1.amazonaws.com/nurego/uaa\_nurego

name: uaa

ports:

- containerPort: 8080

name: uaa

volumeMounts:

- name: config-volume

mountPath: /opt/uaa/config

- name: config-set

mountPath: /mnt

volumes:

- name: config-volume

configMap:

name: uaa.conf

- name: config-set

configMap:

name: uaa.set