

DevOps Sesión 5 (2022-02-23) Swarm

Documentación relacionada

1-Despliegue de Aplicaciones Docker/

Clase

./2-Docker Swarm.pdf pag 17

- modo replicado
 - para no usar modo malla por defecto (solo abre puerto en el lugar (nodo) que se coloca), especificar **mode=host** en **published**
 - `docker service create --replicas 1 --name web4 --publish published=8580,target=80,mode=host dockercloud/hello-world`
- modo global
 - se crea un servicio en cada nodo
 - no permite escalar, ni permite `--replicas`
 - `docker service create --mode global --name servicio4 --limit-memory 128MB --publish 8083:80 httpd`

nodos

- `docker node ...`
 - `demote`: degrada un manager a worker
 - `promote`: promueve un worker a manager
 - `inspect`: información detallada
 - `ls`: listado de nodos
 - `ps`: servicios
 - `rm`: eliminar nodo (`docker swarm leave` para salir del SWARM)
 - `update`: actualizar el estado del nodo
 - `--availability active`: modo normal de trabajo
 - `--availability drain`: deja de recibir tareas, va vaciando las que tiene
 - `--availability pause`: deja de recibir nuevas tareas, mantiene las actuales
 - `--label-add`: añade etiqueta `key=value`
 - `--label-remove`
 - `--role`: worker/manager
- `docker swarm join-token worker`
- https://docs.docker.com/engine/swarm/admin_guide/
- pag 20

etiquetar nodo

- relacionado con **constrains**

- `node.role`
- `node.hostname`
- `node.platform.os`
- se pueden crear etiquetas sobre los nodos para aplicar constrains
 - `docker nodes update --label-add etiqueta=valor`
 - `docker service create --name servicio --constraint 'node.labels.etiqueta==valor'`
 - `docker service create --name servicio --constraint 'node.labels.etiqueta!=valor'`

stacks

```
./Ficheros Laboratorio Stack de servicios swarm/stack-wordpress-swarm
```

- problema de almacenamiento (volúmenes)
- <https://docs.portainer.io/v/ce-2.11/start/install/server/swarm/linux>
 - `curl -L https://downloads.portainer.io/portainer-agent-stack.yml -o portainer-agent-stack.yml`
- networks → attachable: permite que contenedores lanzados fuera del SWARM se puedan añadir a esa red
- `docker stack deploy -c portainer-agent-stack.yml portainer`
- `docker stack ls`
- `docker stack ps portainer`
- `docker service ls`

rolling

cluster

- primera máquina es Master/Worker a la vez
- otros nodos se pueden unir como Master o Worker
 - otros nodos Master no tienen rol worker (no ejecutan contenedores) ← FALSE
 - todos tienen rol Master/Worker, aunque se recomienda que los master solo hagan de master (haciendo un `docker node update --availability drain`)
- mejor 1 Master + 2 Workers que 3 Master/Worker

secret

- `docker secret...`
 - `create`
 - `inspect`
 - `ls`
 - `rm`
- las imágenes han de estar preparadas para usar swarm secrets
 - se añade **_FILE** a las variables de entorno «reales»
 - crea un fichero con la contraseña en plano en `/run/secrets/<secreto>`
- definir los secrets en el servicio y en un apartado específico:

```
version: '3.1'  
services:  
  db:
```

```
image: postgres
restart: always
environment:
  POSTGRES_USER_FILE: /run/secrets/pg_user
  POSTGRES_PASSWORD_FILE: /run/secrets/pg_password
  POSTGRES_DB_FILE: /run/secrets/pg_database
secrets:
  - pg_password
  - pg_user
  - pg_database
adminer:
  image: adminer
ports:
  - 8680:8080
secrets:
  pg_user:
    external: true
  pg_password:
    external: true
  pg_database:
    external: true
```

- `echo «11111111» | docker secret pg_password -`
- `docker secret inspect pg_password`

Extras

- `docker system`
- eliminar contenedores intermedios:
 - NO: <https://nickjanetakis.com/blog/docker-tip-32-automatically-clean-up-after-docker-daily>
 - en cron, `docker prune`
 - <https://docs.docker.com/config/pruning/> → `docker container prune --filter «until=24h»`

From:
<https://miguelangel.torresegea.es/wiki/> - miguel angel torres egea

Permanent link:
<https://miguelangel.torresegea.es/wiki/info:cursos:pue:devops2022:s5>

Last update: **09/03/2022 09:18**

