RECORDAR que els dockers tenen **SO LINUX ALPINE**

**—----------------------------------------------------------------------------------------**

PROBLEMA, hem trobat que localhost resol amb la IPv6, i per això no sap resoldre <https://localhost>. Hem de treure/comentar la línea del /etc/hosts:

**::1  ~~localhost~~  ip6-localhost ip6-loopback**

**—--------------------------------------------------------------------------------**

**DESPLEGAMENT:**

s4 s5 s7 deben despelgarse en el host local(porque son los que comparten archivos (fotos) y base de datos entre ellos)

Recomendado subir todas las imagenes en dockerhub (s1, S2\_3,S4 S5\_6,S7), una vez todo este configurado para que todos los docker cojan de la misma fuente las imagenes y no dependan del "local".

**—----------------------------------------------------------------------------------------------------**

**INCIAR SWARM EN L' EQUIP PRINCIPAL**

| docker swarm init --advertise-addr 10.200.243.224 |
| --- |

**POSAR EL TOKEN GENERAT EN ELS EQUIPS QUE VOLGUEM DELEGAR**

| docker swarm join --token SWMTKN-1-2jzysckb8pjme6nvawwojajtloudtdblt2tofr2couvra7lbkb-d5tda5k5klr3lfwdxye0fte0z 10.200.243.224:2377 |
| --- |

**DESCONETAR/SORTIR DEL SWARM (TANT EL SERV. PRINCIPAL COM EL DELGAT)**

| docker swarm leave --force |
| --- |

**Llistar tots els nodes disponibles dins del SWARM generat**

| docker node ls |
| --- |

| ID HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION szmvpn9jjmfm40gul0sc7w3is i23 Ready Active 20.10.12 86r0ebwaw9rhsqmr6ejefrw0p \* i24 Ready Active Leader 20.10.11 |
| --- |

**Llistar containers i l'estat en què es troben**

| docker stack ps extagram |
| --- |

**Per engegar tots els nodes a tots els hosts que volguem segons labels.**

| -constraints |
| --- |

**Desplegar el swarm (assignar i despelgar els nodes als host) a partir del docker-compose.**

| docker stack deploy --compose-file docker-compose.yml extagram |
| --- |

**Parar tots els nodes.**

| docker stack rm extagram |
| --- |

**Llistar tots els serveis dels nodes del SWARM:**

| docker service ls |
| --- |

**CREAR LABELS:**

*Estan declarades en el docker-compose.yml (a l'opció constraints)*

| docker node update --label-add type=storage edveaz4keymoh728inrhu1tio edveaz4keymoh728inrhu1tio |
| --- |

**----------------------------------------------------**

**TEORIA DE PORTS:**

TENIM UN HOST CENTRALITZANT (AQUEST ON ES MONTARA AMB DOCKER -COMPOSE TOTS ELS docker) obviamente no pot reptir ports.

Cada docker pot obrir els ports que vulgui ja que són independents

**DOCKER COMPOSE.yml** SURTEN ELS PORTS QUE S'EXPOSARAN (SERAN ACCESIBLES ) PER AL EXTERIOR. A MÉS TAMBÉ PODEM MAPEJAR EL PORT A UN ALTRE PORT LLIURE (Ex **22:2200**) I EXPOSAR EL SERVEI CAP AL NOU PORT (SOBRETOT ES FA PER NO TENIR CONFLICTES DE SOLAPAMENT DE PORTS )

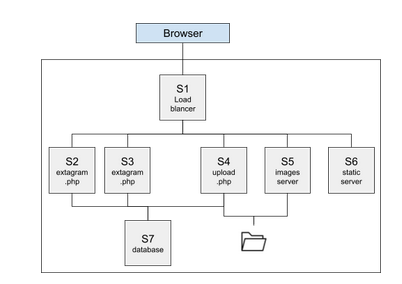
**DOCKERFILE** TENIM EL **EXPOSE:** ON INDIQUEM QUE EL DOCKER EN CUESTIÓ EXPOSARA **DINS LA XARXA DOCKER** ELS PORTS INDICATS.

En el cas concret de **l'activitat extagram :**

-En els docker file NO cal exposar els ports.

**DESPELGAMENT:**

~~-Es important l'ordre que anem despelgant els docker ja que uns depenen dels altres (veure dibuix). AIXO ESTA EN EL "DEPENDS ON" DEL~~ **~~docker\_compose.yml~~** (deprecated)



**docker-compose up -d - -build** → Aixeca fent build a totes les imatges implicades

**docker-compose ps** → Veure estat dels docker aixecats

**docker-compose logs [<server\_name>]** → Veure errors

Per a agafar el arxius de configuració :

Posem dins del fitxer DOCKER-COMPOSE.yml:

Dins del docker:

S6:

configs:

source: **s6\_nginx\_conf**

target: /etc/nginx/conf.d/default.conf

Fora dels docker (alfinal del fitxer DOCKER-COMPOSE.yml:)

configs:

**s6\_nginx\_conf**

file: s6/defaults.conf ← Directori dels fitxers del host que executa el docker-compose

--------------------------------------------

Dins del docker:

aliases: => Otros nombre para referenciar al servidor (utilitzem per el mysql)

------------------------------------------

Dins del docker:

volumes (és dirà mount) = per montar i no tenir dins dels dockers o les imatges certs fitxers (ja que sinó al fer down es borrarien del docker).

volumes:

- ./dbdata:/var/lib/mysql

Per asignar valor a les variables:

environment:

MYSQL\_ROOT\_PASSWORD: password

MYSQL\_DATABASE: extagram\_db

MYSQL\_USER: extagram\_admin

MYSQL\_PASSWORD: pass12

Encriptar un arxiu en questió (normalment keys)

(fora dels docker)

secrets:

nginx\_key:

file: s1/site.key