

SAY HI TO VIRTUALIZATION

DOCKER SWARM

Wang Xuechen

2019/09/10

AGENDA

- ▶ Docker Swarm
 - ▶ Architecture
 - ▶ Ingress network
 - ▶ Image/Container/Service/Stack/Swarm
- ▶ How-to
 - ▶ compose-file
 - ▶ config / volumes
 - ▶ network drivers

DOCKER SWARM

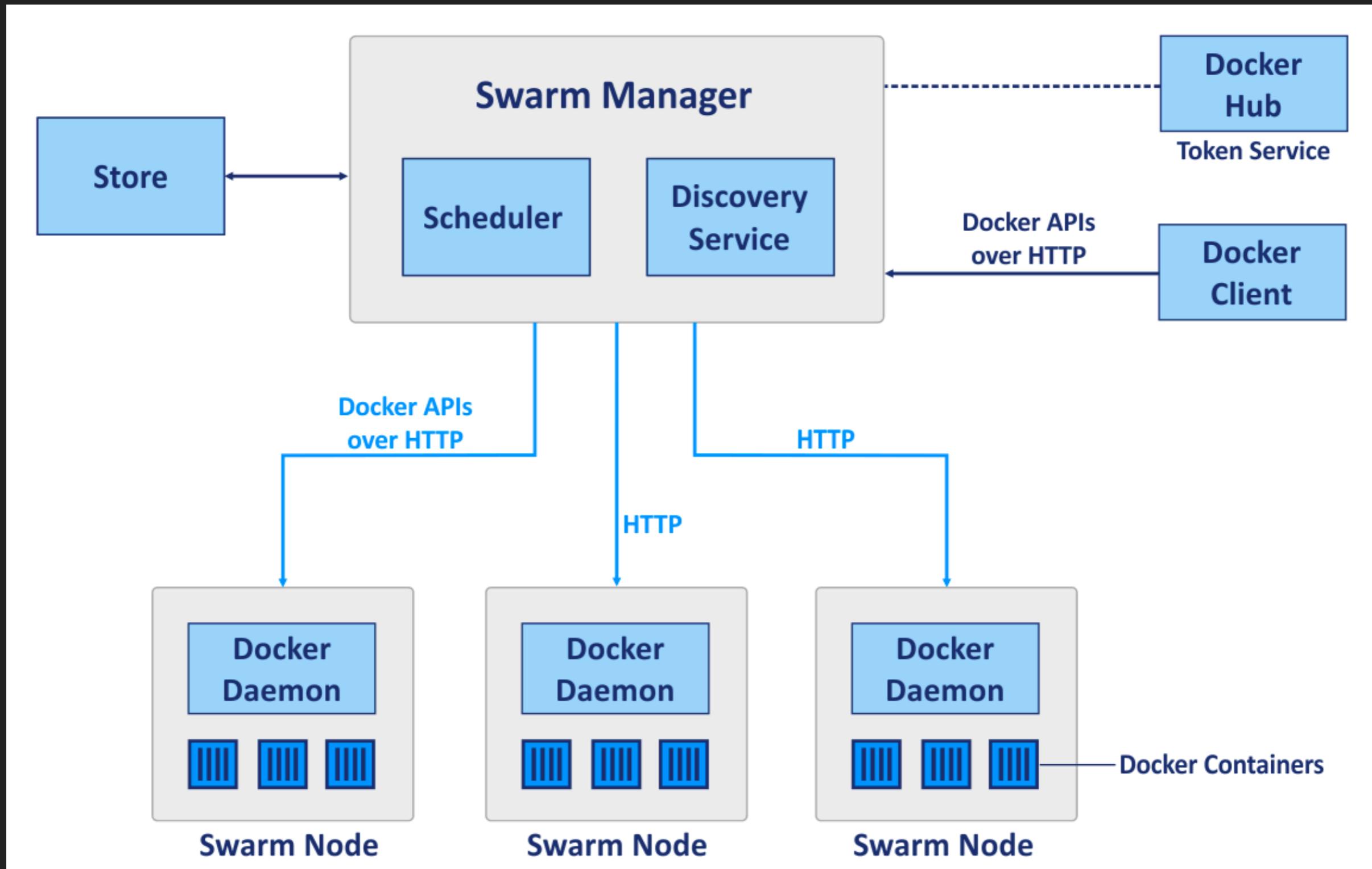


MESOS

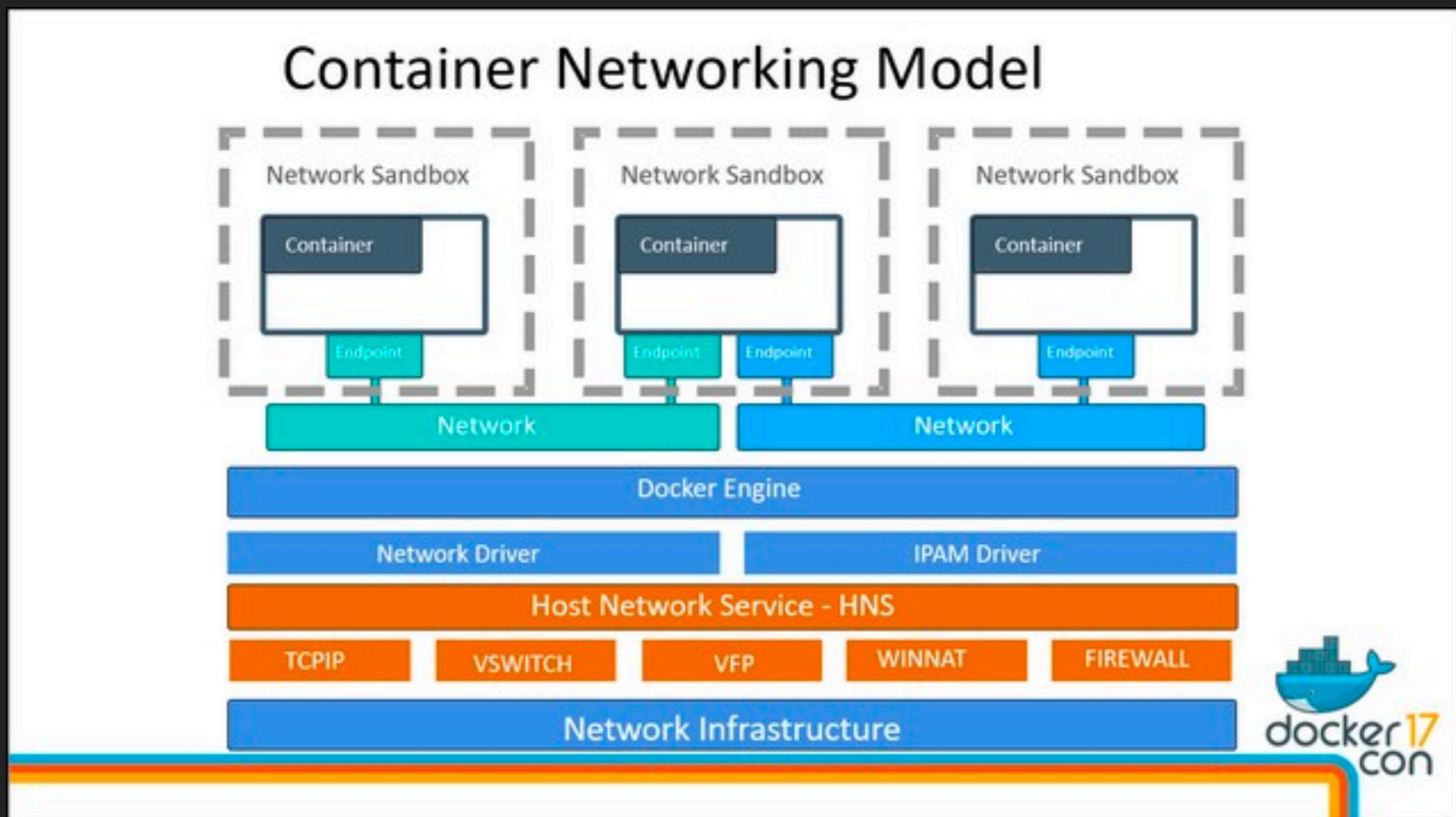
DOCKER SWARM

- ▶ <https://docs.docker.com/engine/swarm/>
 - ▶ Cluster management integrated with Docker Engine
 - ▶ Decentralized design
 - ▶ Declarative service model
 - ▶ Scaling
 - ▶ Desired state reconciliation
 - ▶ Multi-host networking
 - ▶ Service discovery
 - ▶ Load balancing
 - ▶ Secure by default
 - ▶ Rolling updates

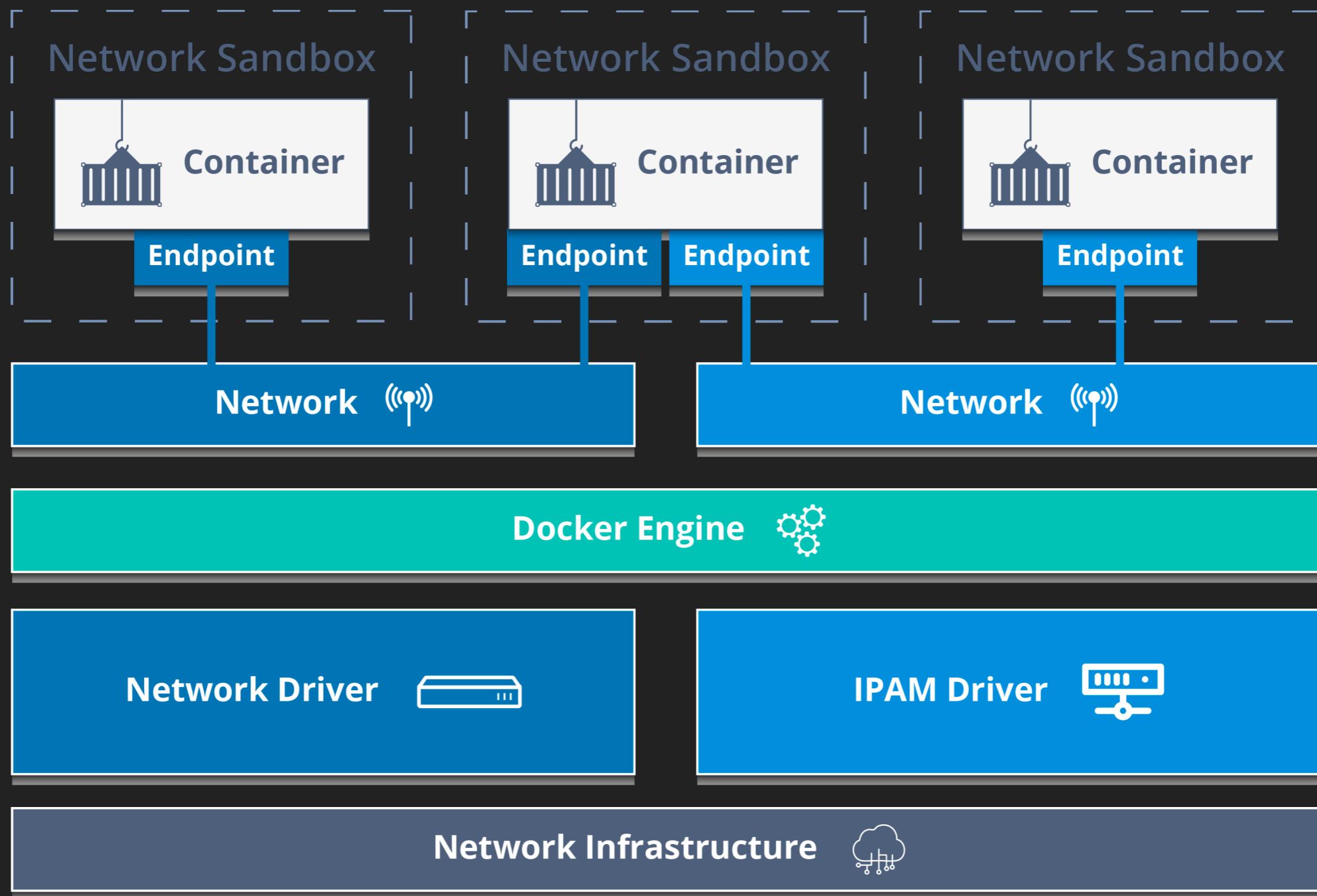
DOCKER SWARM



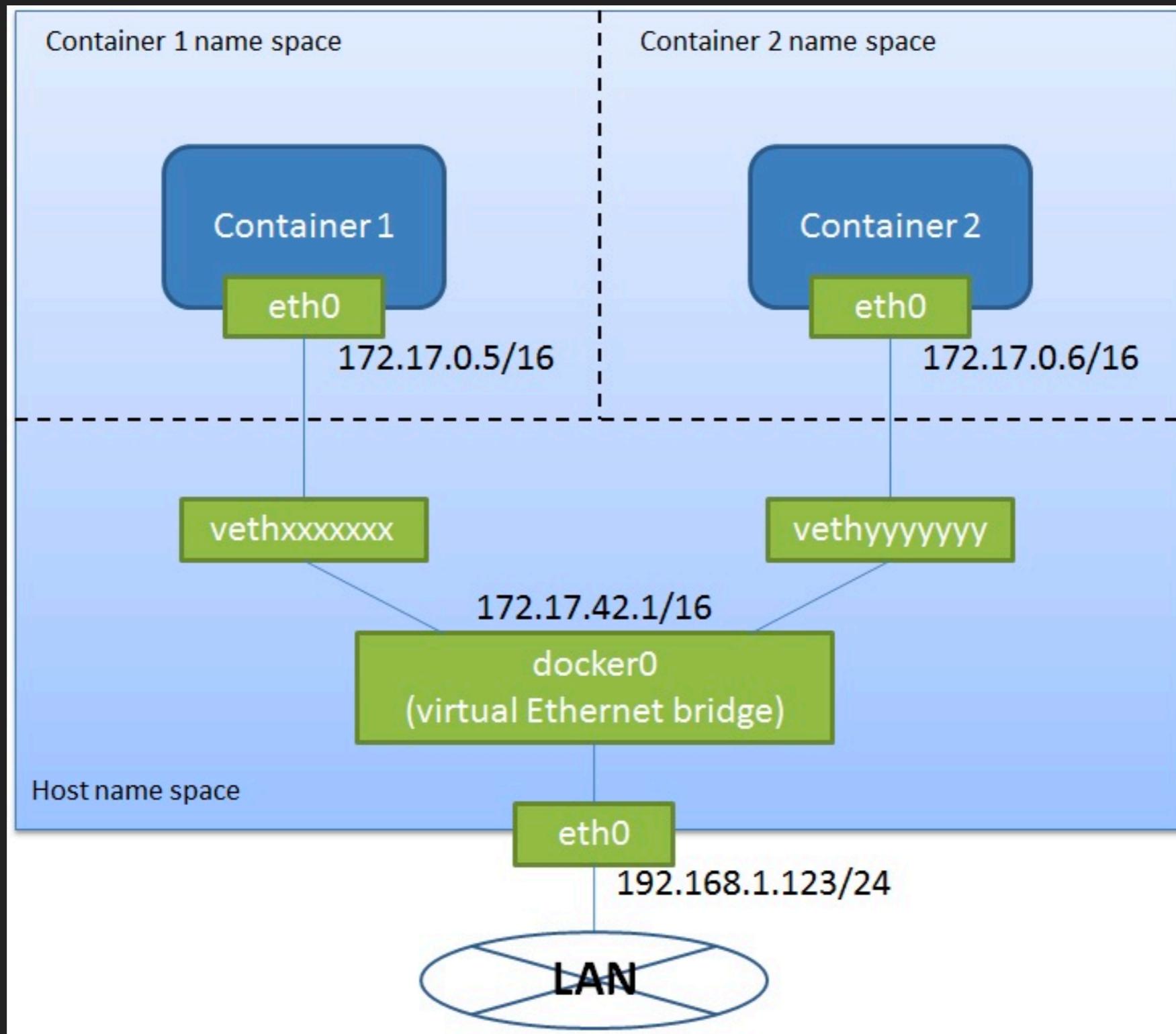
DOCKER NETWORKING



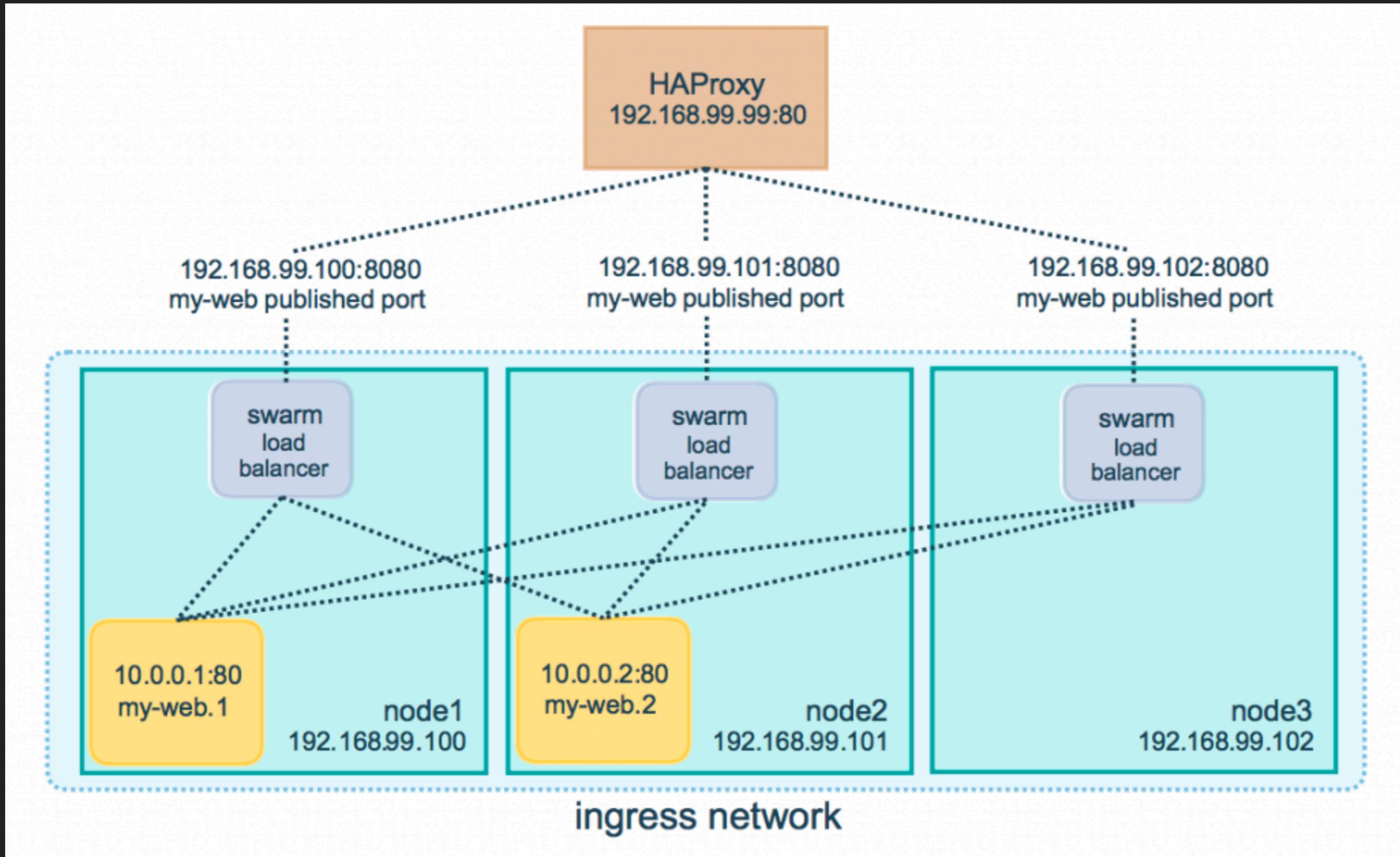
DOCKER NETWORKING



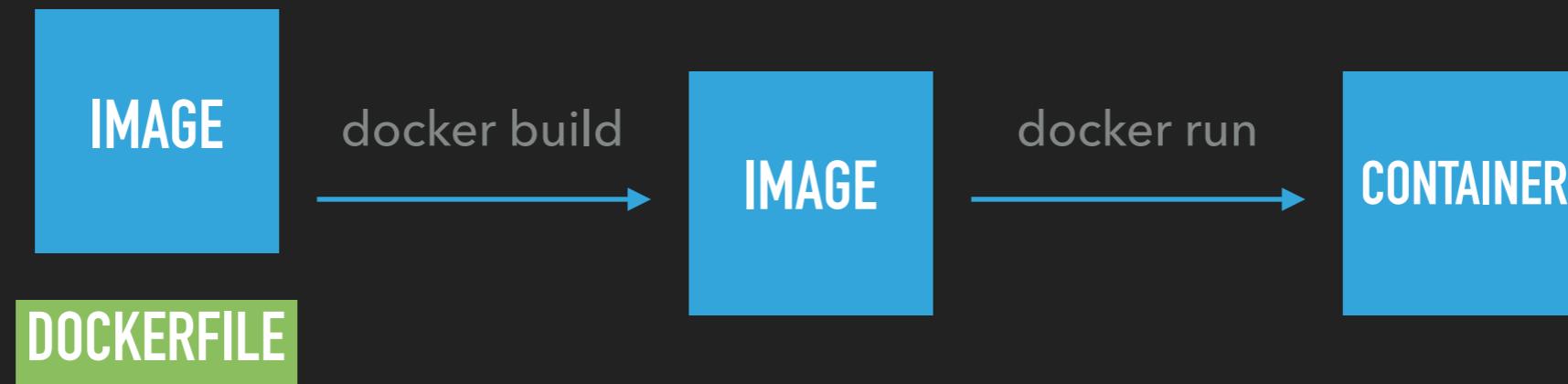
DOCKER NETWORKING



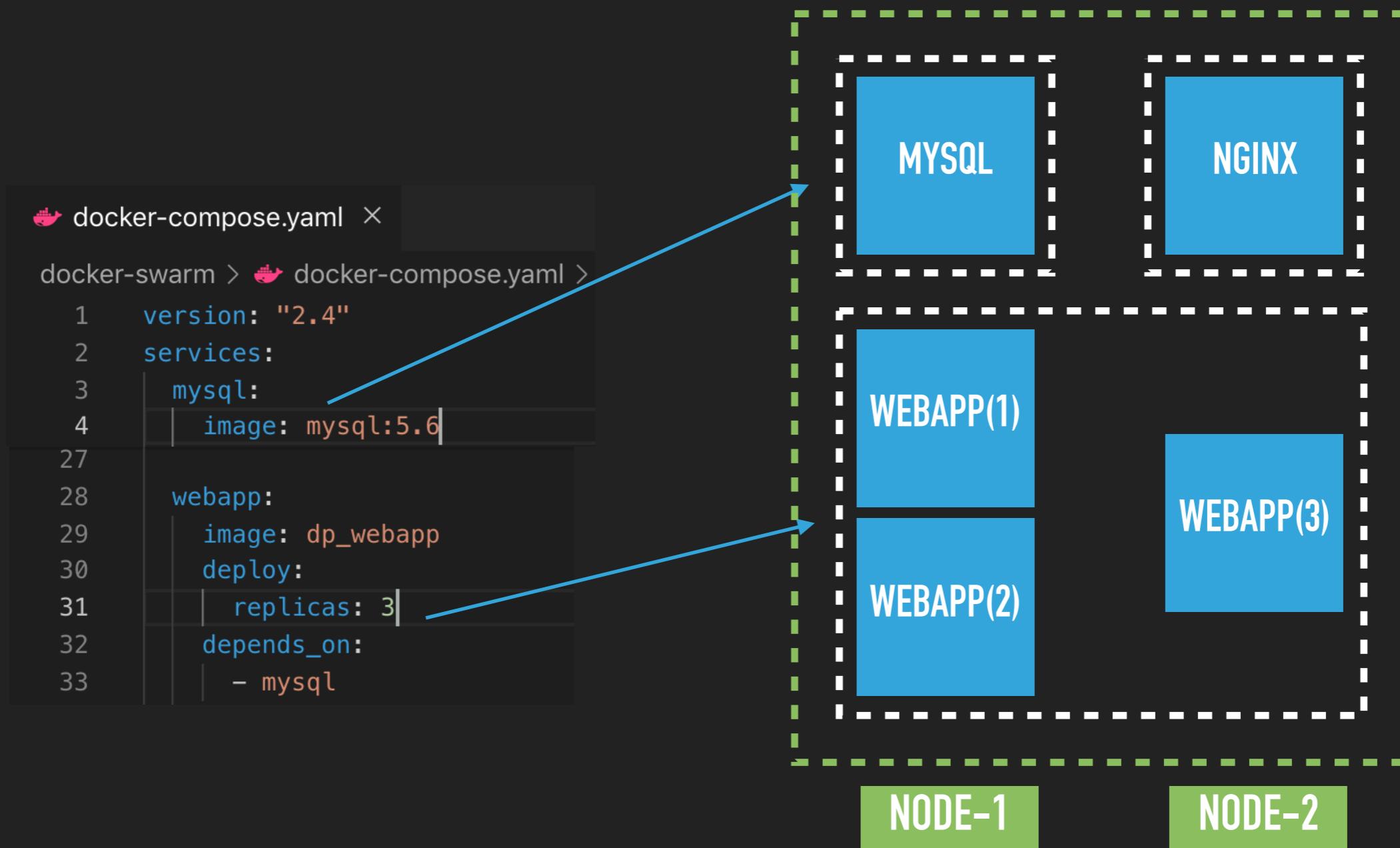
DOCKER SWARM



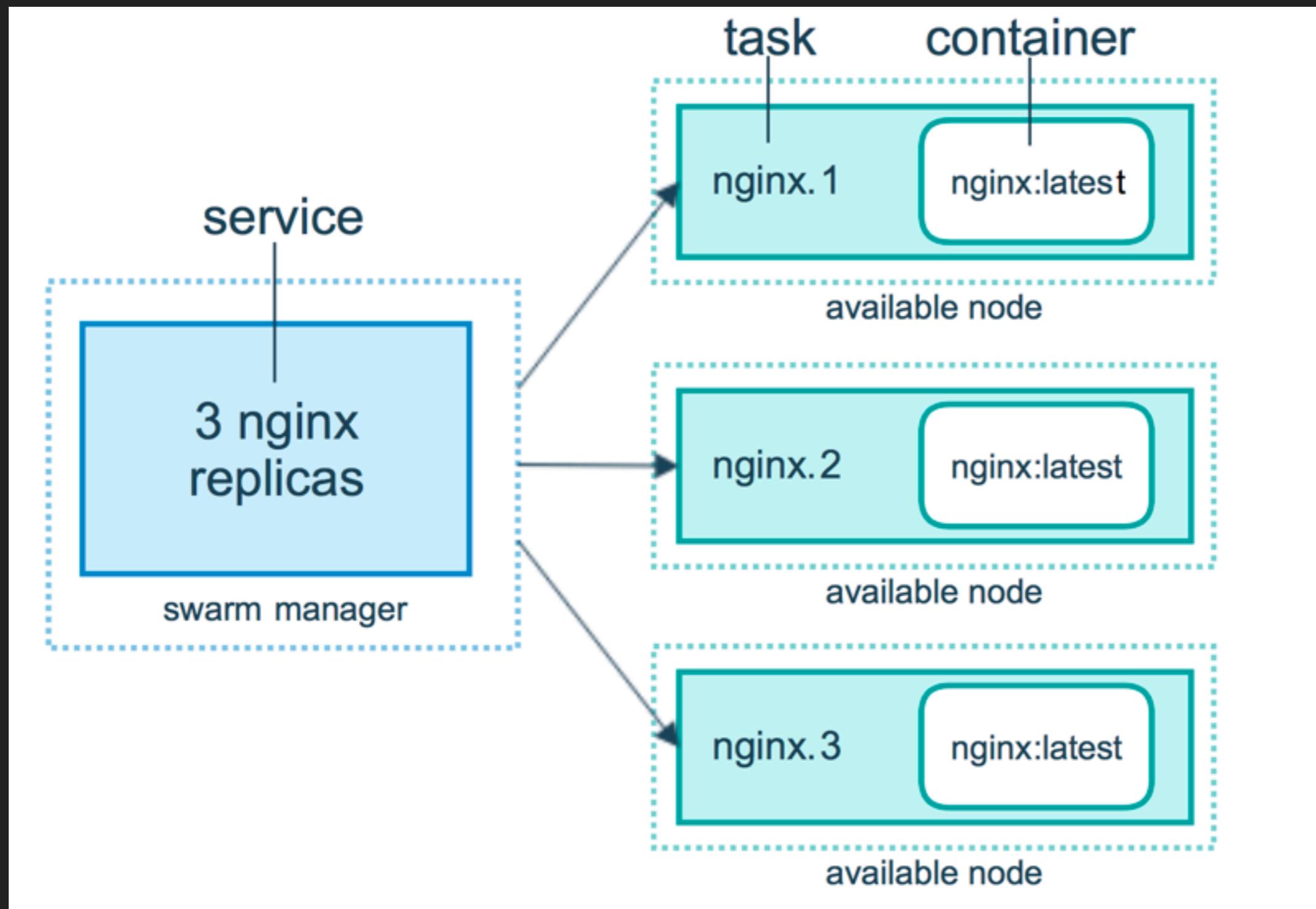
IMAGE/CONTAINER



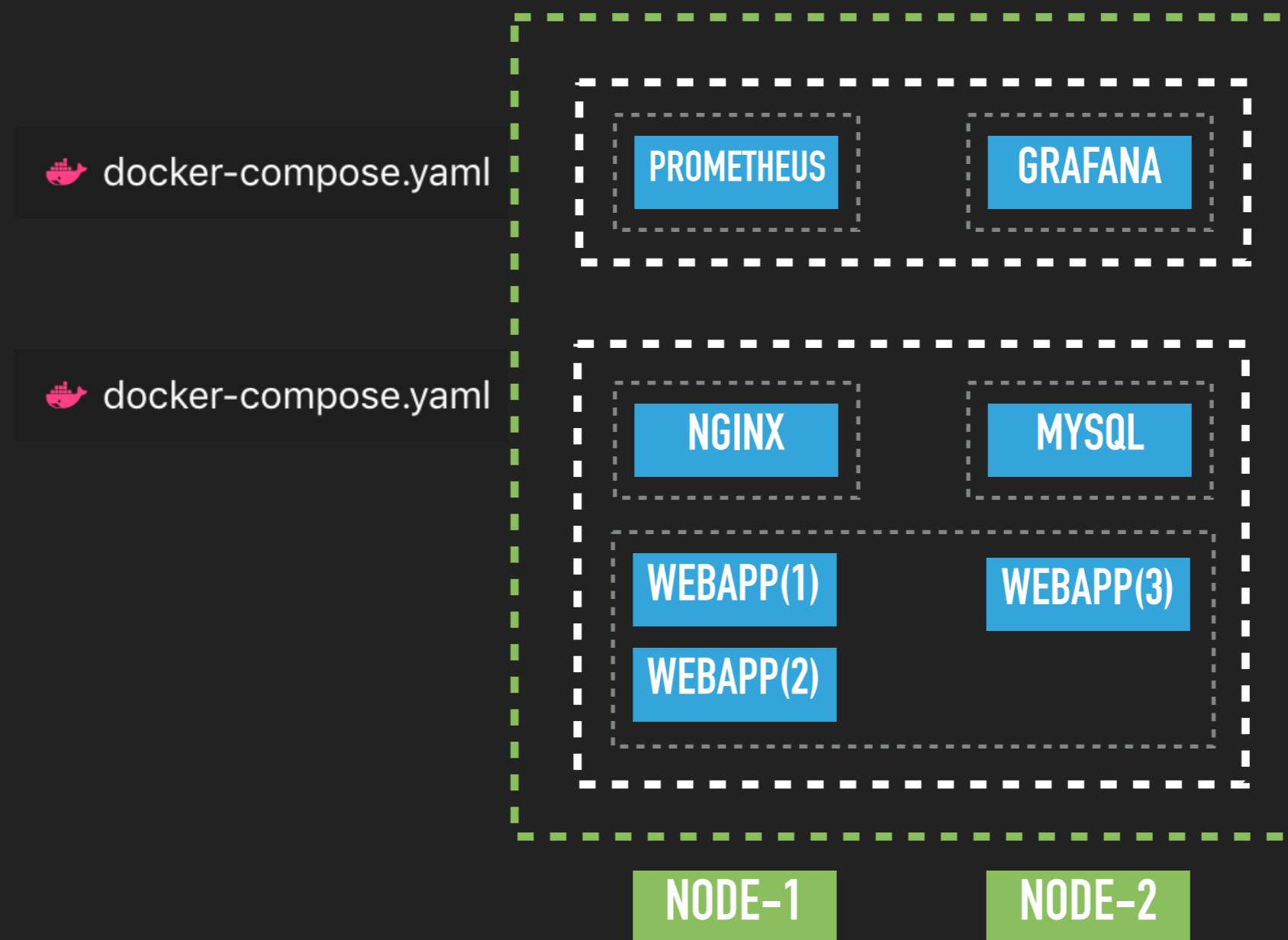
SWARM - SERVICE



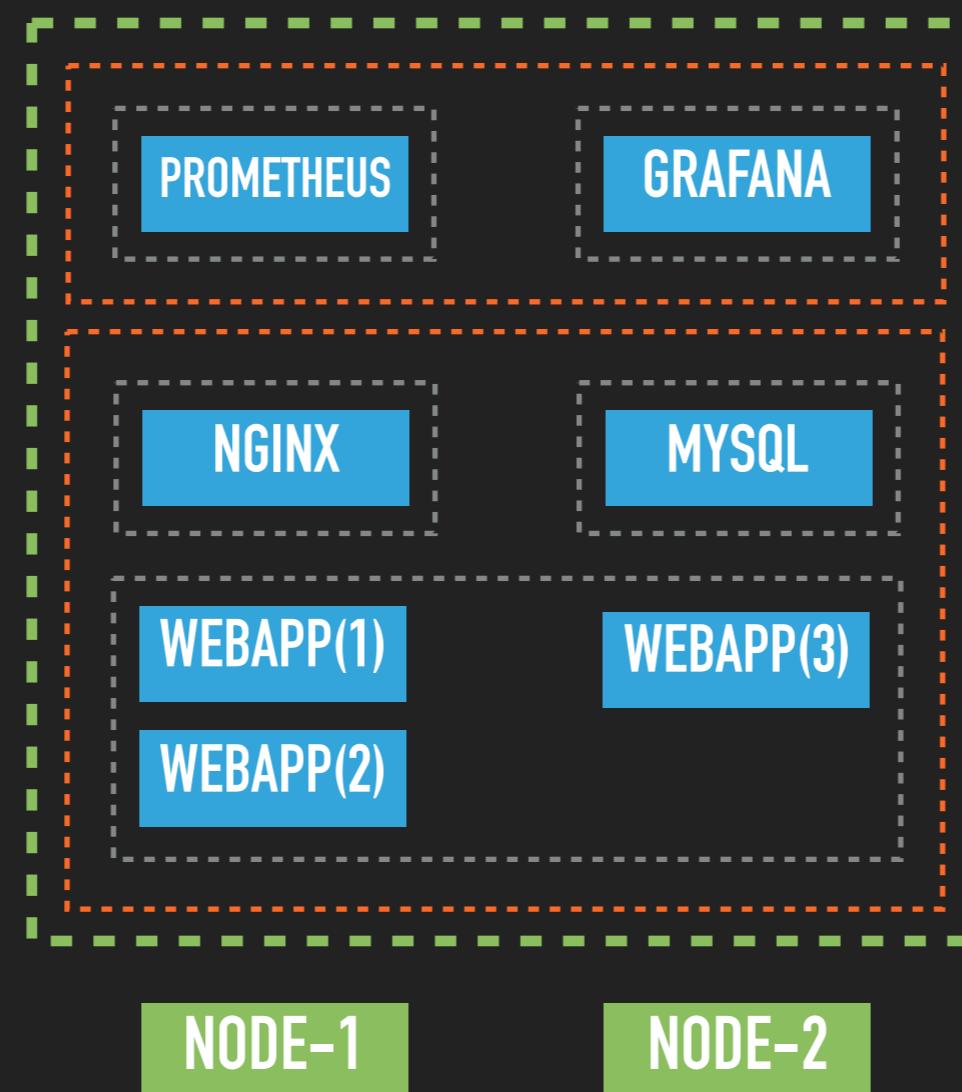
SWARM - SERVICE



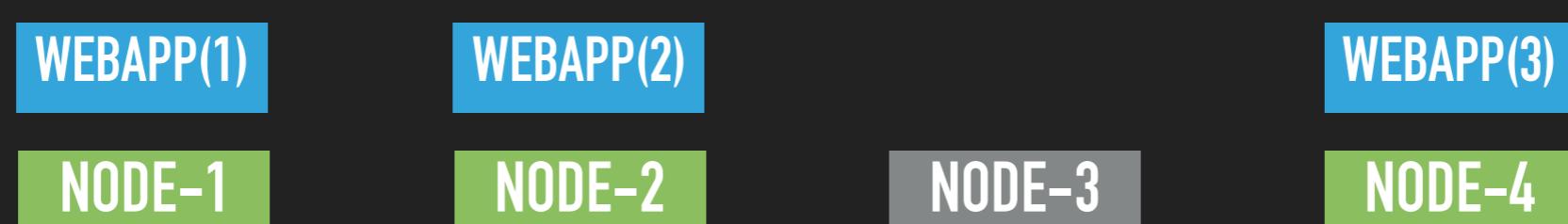
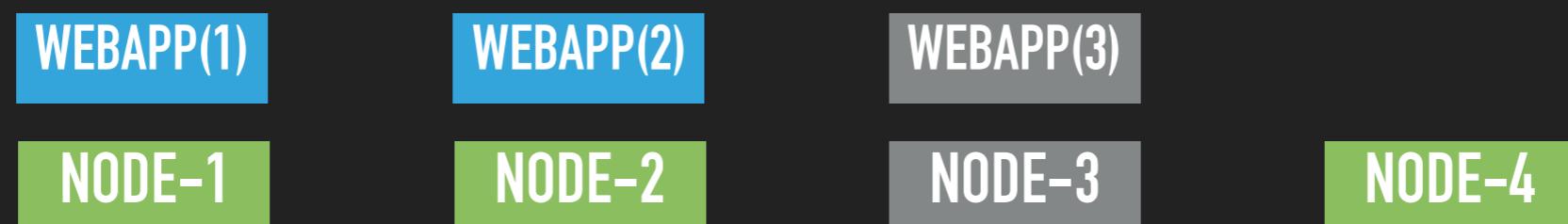
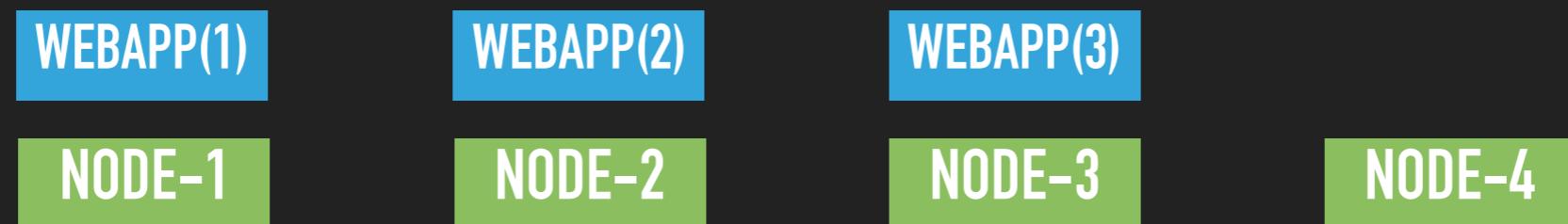
SWARM - STACK



SWARM



SWARM



COMPOSE FILE



https://github.com/zenanswer/docker_practice/blob/master/docker-swarm/docker-compose.yaml

COMPOSE FILE

```
ESH: docker-compose.yaml ×  
docker-swarm > ESH: docker-compose.yaml > {} ser  
77  
78   configs:  
79     healthcheck_js:  
80       file: ./docker/healthcheck.js  
81     nginx_conf:  
82       file: ./docker/nginx.conf  
83     # one_external_config:  
84     #   external: true  
85  
86   # secrets:  
87   #   one_external_secret:  
88   #     external: true  
89  
90   volumes:  
91     app-img-data:  
92  
93   networks:  
94     app_network:  
95       driver: overlay  
96       attachable: true  
97
```

docker > ⚡ docker-compose.yaml > abc version

```
1 -version: "3.7"
2 -
3   services:
4     mysql:
5       image: mysql:5.6
6       deploy:
7         restart_policy:
8           condition: on-failure
9         placement:
10          constraints: [node.role == manager]
11       volumes:
12         - type: bind
13           source: ./data/mysql
14           target: /var/lib/mysql
15         - type: bind
16           source: ../webapp/node-mysql-crud-app/create_table.
17           target: /docker-entrypoint-initdb.d/create_table.sq
18         - type: bind
19           source: ../docker/checkiftableexists.sh
20           target: /checkiftableexists.sh
21   environment:
22     - MYSQL_ROOT_PASSWORD=root
23     - MYSQL_USER=dpuser
24     - MYSQL_PASSWORD=dppassed
25     - MYSQL_DATABASE=dp
26   healthcheck:
27     test: ["CMD", "bash", "/checkiftableexists.sh", "dp",
28     start_period: 30s
29     interval: 5s
30     timeout: 3s
31     retries: 4
32   networks:
33     - app_network
34
```

```
1 +version: "2.4"
2   services:
3     mysql:
4       image: mysql:5.6
5       restart: on-failure
6   volumes:
7     - type: bind
8       source: ./data/mysql
9       target: /var/lib/mysql
10    - type: bind
11      source: ../webapp/node-mysql-crud-app/create_table.
12      target: /docker-entrypoint-initdb.d/create_table.s
13    - type: bind
14      source: ./checkiftableexists.sh
15      target: /checkiftableexists.sh
16   environment:
17     - MYSQL_ROOT_PASSWORD=root
18     - MYSQL_USER=dpuser
19     - MYSQL_PASSWORD=dppassed
20     - MYSQL_DATABASE=dp
21   healthcheck:
22     test: ["CMD", "bash", "/checkiftableexists.sh", "dp",
23     start_period: 30s
24     interval: 5s
25     timeout: 3s
26     retries: 4
```

DOCKER - HOW-TO

COMPOSE FILE

```
gist.github.com/abc/docker-compose.diff
```

docker > docker-compose.yaml > abc version

Line	Original	Modified
62	nginx:	nginx:
63	image: nginx:1.17.3	image: nginx:1.17.3
64	ports:	ports:
65	- "80:80"	- "80:80"
66	- deploy:	
67	restart_policy:	restart: on-failure
68	condition: on-failure	
69	- configs:	volumes:
70	- source: nginx_conf	- type: bind
71	target: /etc/nginx/nginx.conf	source: ./nginx.conf
72	mode: 0440	target: /etc/nginx/nginx.conf
73	depends_on:	depends_on:
74	- webapp	webapp:
75	- networks:	condition: service_healthy
76	- app_network	

SWARM - INIT & JOIN

► docker swarm/node status

```
[xcwang@alidev project]$ docker info | grep -i swarm
```

Swarm: inactive

```
[xcwang@alidev project]$
```

Run "docker swarm init" or "docker swarm join" to connect this node to swarm and try again.

```
[xcwang@tag01 ~]$ docker info | grep -i swarm
```

Swarm: inactive

```
[xcwang@tag01 ~]$
```

```
[xcwang@alidev project]$ docker node ls
```

Error response from daemon: This node is not a swarm manager. Use "docker swarm init" or "docker swarm join" to connect this node to swarm and try again.

```
[xcwang@alidev project]$
```

```
[xcwang@tag01 ~]$ docker node ls
```

Error response from daemon: This node is not a swarm manager. Use "docker swarm init" or "docker swarm join" to connect this node to swarm and try again.

```
[xcwang@tag01 ~]$
```

SWARM - INIT & JOIN

► docker swarm init & node join

```
[xcwang@alidev project]$ docker swarm init
Swarm initialized: current node (1uytgicfqnl4hhmht0r8a62dk) is now a manager.
```

To add a worker to this swarm, run the following command:

```
docker swarm join --token SWMTKN-1-210sfq0pfyrrnq59azzdxvgdapb7fmedzxvz1f270nym6yxdim4-aowmtg7y1eva2ljq0uwewgkft 10.10.11.5:2377
```

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

```
[xcwang@alidev project]$
```

```
[xcwang@tag01 ~]$ docker swarm join --token SWMTKN-1-210sfq0pfyrrnq59azzdxvgdapb7fmedzxvz1f270nym6yxdim4-aowmtg7y1eva2ljq0uwewgkft 10.10.11.5:2377
This node joined a swarm as a worker.
[xcwang@tag01 ~]$
```

SWARM - INIT & JOIN

► docker swarm/node status

```
[xcwang@alidev project]$ docker info | grep -i swarm
```

Swarm: active

```
[xcwang@alidev project]$ █
```

```
[xcwang@tag01 ~]$
```

```
[xcwang@tag01 ~]$ docker info | grep -i swarm
```

Swarm: active

```
[xcwang@tag01 ~]$ █
```

```
[xcwang@alidev project]$ docker node ls
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGE
R STATUS	ENGINE VERSION			
1uytgicfqnl4hhmht0r8a62dk *	alidev	Ready	Active	Leader
	19.03.1			
1qq5shhn0ywb8enhfbfioid6c	tag01	Ready	Active	
	19.03.1			

```
[xcwang@alidev project]$
```

```
[xcwang@alidev project]$
```

```
[xcwang@alidev project]$ █
```

```
[xcwang@tag01 ~]$ docker node ls
```

Error response from daemon: This node is not a swarm manager. Worker nodes can't be used to view or modify cluster state. Please run this command on a manager node or promote the current node to a manager.

```
[xcwang@tag01 ~]$ █
```

SWARM - INIT & JOIN

► docker swarm/node status

```
Log: awslogs fluentd gcplogs gelf journald json-file syslog
Swarm: active
NodeID: 1uytgicfqnl4hhmht0r8a62dk
Is Manager: true
ClusterID: gng25wv9or8uw2lie2lxwc5bx
Managers: 1
Nodes: 2
Default Address Pool: 10.0.0.0/8
SubnetSize: 24
Data Path Port: 4789
Orchestration:
  Task History Retention Limit: 5
Raft:
```

```
Network: bridge host ipvlan macvlan null overlay
Log: awslogs fluentd gcplogs gelf journald json-file syslog
Swarm: active
NodeID: lqq5shhn0ywb8enhfbfioid6c
Is Manager: false
Node Address: 10.10.11.6
Manager Addresses:
  10.10.11.5:2377
Runtimes: runc
Default Runtime: runc
Init Binary: docker-init
```

SWARM - CUSTOM IMAGE

- ▶ dp_webapp image
- ▶ [https://github.com/zenanswer/docker_practice/tree/
master/webapp](https://github.com/zenanswer/docker_practice/tree/master/webapp)
- ▶ Make sure that every node in this swarm has this image.

SWARM - STACK

► docker stack deploy

```
[xcwang@alidev docker-swarm]$ docker stack deploy --compose-file ./docker-compose.yaml dptest
Creating network dptest_app_network
Creating config dptest_nginx_conf
Creating config dptest_healthcheck_js
Creating service dptest_nginx
Creating service dptest_mysql
Creating service dptest_webapp
```

```
[xcwang@alidev docker-swarm]$ docker stack ls
NAME          SERVICES          ORCHESTRATOR
dptest        3                Swarm
```

SWARM - STACK

- ▶ docker stack deploy [error]

ID	NAME	IMAGE	NODE	DESIRED STATE
CURRENT STATE		ERROR		PORTS
t6kwn7qi2b8j	dptest_webapp.1	dp_webapp:latest	tag01	Ready
	Preparing 3 seconds ago			
zn8um69cyw0a	_ dptest_webapp.1	dp_webapp:latest	tag01	Shutdown
	Rejected 4 seconds ago	"No such image: dp_webapp:late..."		
x35lna6abbvf	dptest_mysql.1	mysql:5.6	alidev	Running
	Preparing 48 seconds ago			
qns4rrzfnprg	dptest_nginx.1	nginx:1.17.3	tag01	Running
	Preparing about a minute ago			
2ex849jq9l0t	dptest_webapp.2	dp_webapp:latest	alidev	Running
	Preparing 25 seconds ago			
xut588z2pphd	dptest_webapp.3	dp_webapp:latest	alidev	Running
	Starting 3 seconds ago			

SWARM - STACK

► docker stack deploy [error]

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE
ERROR			PORTS		
p9qg7aeklwu9tst1tcnqwpaxr	dptest_mysql.1	mysql:5.6@sha256:e2388e706b4e90b2f558126f98eda			
5b38fc36c9d220468a10535778e28707e2f	alidev			Running	Running 33 seconds ago
sxo51gb8r0c2vvo1kdi3yn8kf	dptest_webapp.1	dp_webapp:latest	alidev	Running	Running 29 seconds ago
bh7s4ucnptjklva0taijemilp	dptest_nginx.1	nginx:1.17.3@sha256:53ddb41e46de3d63376579acf4			
6f9a41a8d7de33645db47a486de9769201fec9	tag01			Running	Preparing about a minute ago
id1er3a4776xx2u289dd5ibxv	dptest_webapp.1	dp_webapp:latest	alidev	Shutdown	Failed 53 seconds ago
	"task: non-zero exit (1)"				
kdyfg27eu6xaa5xjzhjb6fxpa	dptest_webapp.2	dp_webapp:latest	alidev	Running	Preparing 11 seconds ago
2juefrfv3dc3olkz9jvq7354p	_ dptest_webapp.2	dp_webapp:latest	tag01	Shutdown	Rejected 11 seconds ago
	"No command specified"				

SWARM - STACK

- ▶ docker stack deploy [error]

```
[xcwang@alidev docker-swarm]$ docker config ls
ID                      NAME          CREATED        UPDATED
pyvzx16zzpzjrvtsitys4949h  dptest_healthcheck_js  7 minutes ago  7 minutes ago
1mpiz88yt0ssxivi2g695op1m   dptest_nginx_conf      7 minutes ago  7 minutes ago
[xcwang@alidev docker-swarm]$ docker network ls
NETWORK ID          NAME          DRIVER        SCOPE
z01za4duf9eu        app_app_network  overlay       swarm
92f1c65415a2        bridge         bridge       local
e44a5472c671        docker_gwbridge  bridge       local
o04jg124lggs        dptest_app_network  overlay       swarm
b22905a890f0        host           host        local
pjptwirwsgb0        ingress        overlay       swarm
897e26a66be2        none          null        local
```

SWARM - STACK

► docker stack deploy [error]

```
[xcwang@alidev docker-swarm]$ docker config inspect dptest_healthcheck_js
[{"id": "pyvzx16zzpzjrvtsitys4949h", "version": {"index": 2099}, "createdAt": "2019-09-09T06:55:35.947197047Z", "updatedAt": "2019-09-09T06:55:35.947197047Z", "spec": {"name": "dptest_healthcheck_js", "labels": {"com.docker.stack.namespace": "dptest"}, "data": "Y29uc3QgaHR0cCA9IHJlcXVpcmUoJ2h0dHAnKTsKCnZhciBwb3J0ID1BPUIQ7Cn0KCmNvbnN0IG9wdGlvbnMgPSB7CiAgICBob3N00iAnMTI3LjAuMC4xJywKICAgIHBydWVzdChvcHRpb25zLCocmVzKSA9PiB7CiAgICBjb25zb2x1LmxvZyhgSEVBTFRIQ0hFQ0sgU1wogICAgICAgIHByb2Nlc3MuZXhpdcgwKTsKICAgIH0KICAgIGVsc2UgewogICAgICAgIHByb2NyKSB7CiAgICBjb25zb2x1LmVycm9yKCdFU1JPUicp0wogICAgchJvY2Vzcy5leGl0KDEp0wp9KT"}]}
```

SWARM - STACK

► docker stack deploy

```
[xcwang@alidev docker-swarm]$ docker stack deploy -c docker-compose.yaml dptest
Creating network dptest_app_network
Creating config dptest_healthcheck_js
Creating config dptest_nginx_conf
Creating service dptest_mysql
Creating service dptest_webapp
Creating service dptest_nginx
[xcwang@alidev docker-swarm]$ █
```

[xcwang@alidev docker-swarm]\$ docker stack ps dptest				
ID	DESIRED STATE	NAME	IMAGE	NODE
		CURRENT STATE	ERROR	PORTS
otrdmblqlqhd	Running	dptest_nginx.1	nginx:1.17.3	tag01
gag235xiy1m7	Running	dptest_webapp.1	dp_webapp:latest	tag01
tuyp4ed7cud9	Running	dptest_mysql.1	mysql:5.6	alidev
du7abbnmk1e7	Running	dptest_webapp.2	dp_webapp:latest	tag01
gkkypaw9ityh	Running	dptest_webapp.3	dp_webapp:latest	alidev

```
[xcwang@alidev docker-swarm]$ █
```

SWARM - STACK

► docker stack deploy

```
[xcwang@alidev docker-swarm]$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
95a7d53256e1	dp_webapp:latest	"node app.js"	2 minutes ago
Up 2 minutes (healthy)	80/tcp	dptest_webapp.3.gkkypaw9ityhvc98	1g761kv14
99352bdfb713	mysql:5.6	"docker-entrypoint.s..."	3 minutes ago
Up 3 minutes (healthy)	3306/tcp	dptest_mysql.1.tuyp4ed7cud9cwa6r	ilddmqyl

```
[xcwang@alidev docker-swarm]$ █
```

```
[xcwang@tag01 ~]$
```

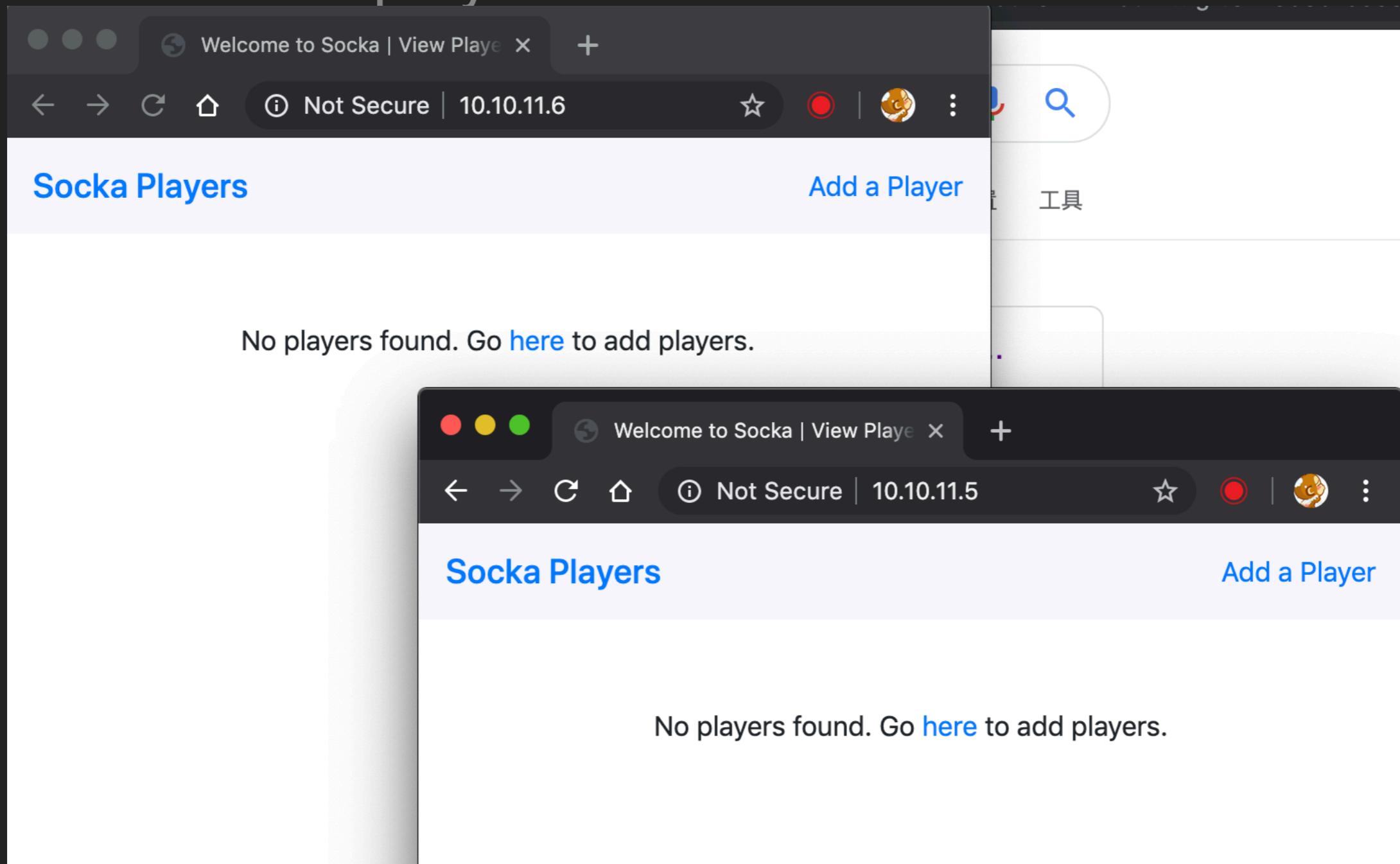
```
[xcwang@tag01 ~]$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STA
TUS	PORTS	NAMES		
b77c4ee98f29	nginx:1.17.3	"nginx -g 'daemon of..."	2 minutes ago	Up
2 minutes	80/tcp	dptest_nginx.1.otrdmbiplqhd4s6h540v2esxg		
1682ff1e2297	dp_webapp:latest	"node app.js"	3 minutes ago	Up
3 minutes (healthy)	80/tcp	dptest_webapp.1.gag235xiy1m7t6x7a5p1vzv99		
7711f5d09b43	dp_webapp:latest	"node app.js"	3 minutes ago	Up
3 minutes (healthy)	80/tcp	dptest_webapp.2.du7abbnmk1e7lcx9w4ljjz9z38		

```
[xcwang@tag01 ~]$ █
```

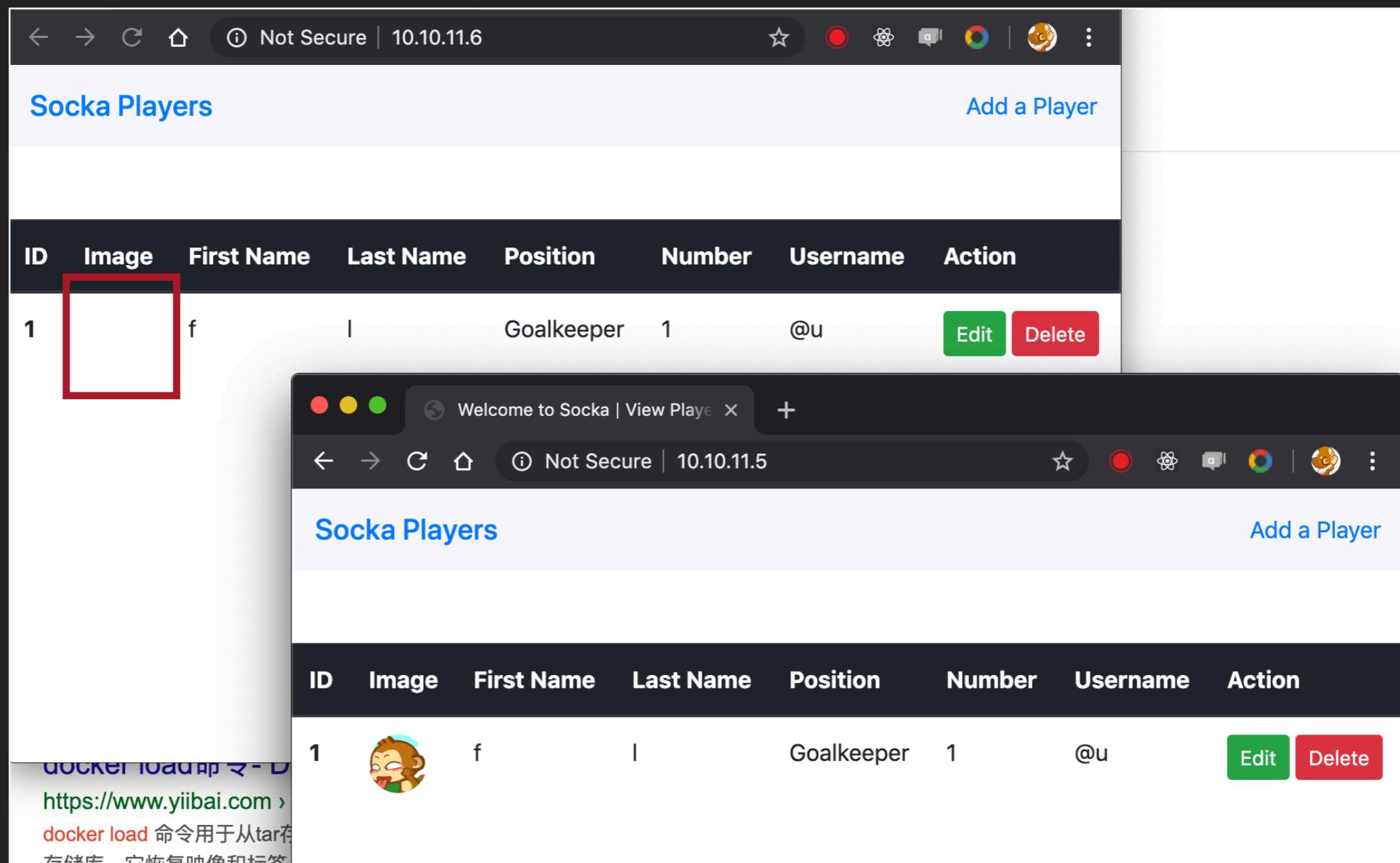
SWARM - STACK

► docker stack deploy



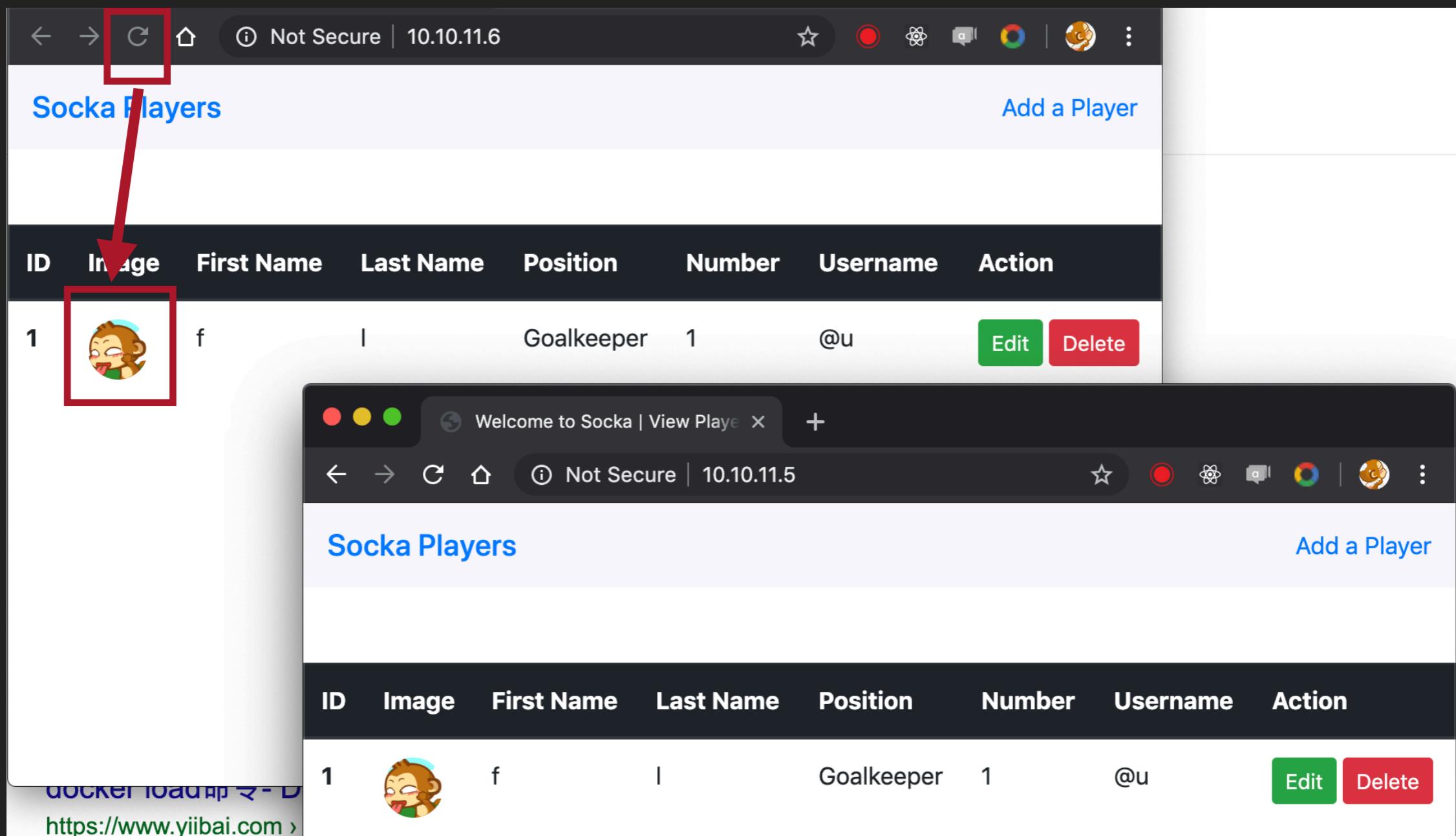
SWARM - STACK

► docker stack deploy



SWARM - STACK

▶ docker stack deploy



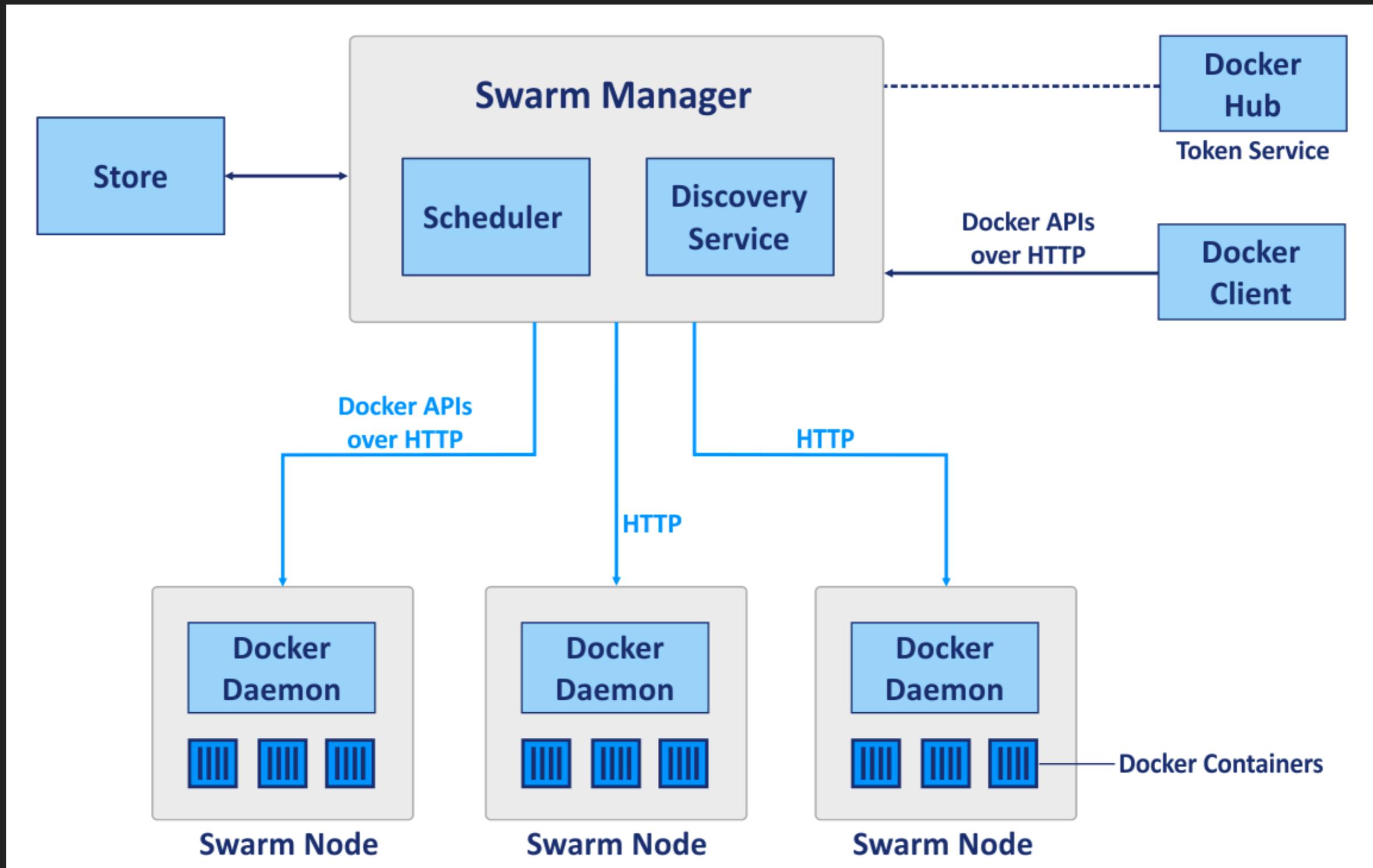
VOLUME

```
[xcwang@alidev docker-swarm]$ docker volume inspect dptest_app-img-data
[
  {
    "CreatedAt": "2019-09-09T16:17:40+08:00",
    "Driver": "local",
    "Labels": {
      "com.docker.stack.namespace": "dptest"
    },
    "Mountpoint": "/home/dockerstore/volumes/dptest_app-img-data/_data",
    "Name": "dptest_app-img-data",
    "Options": null,
    "Scope": "local"
  }
]
```

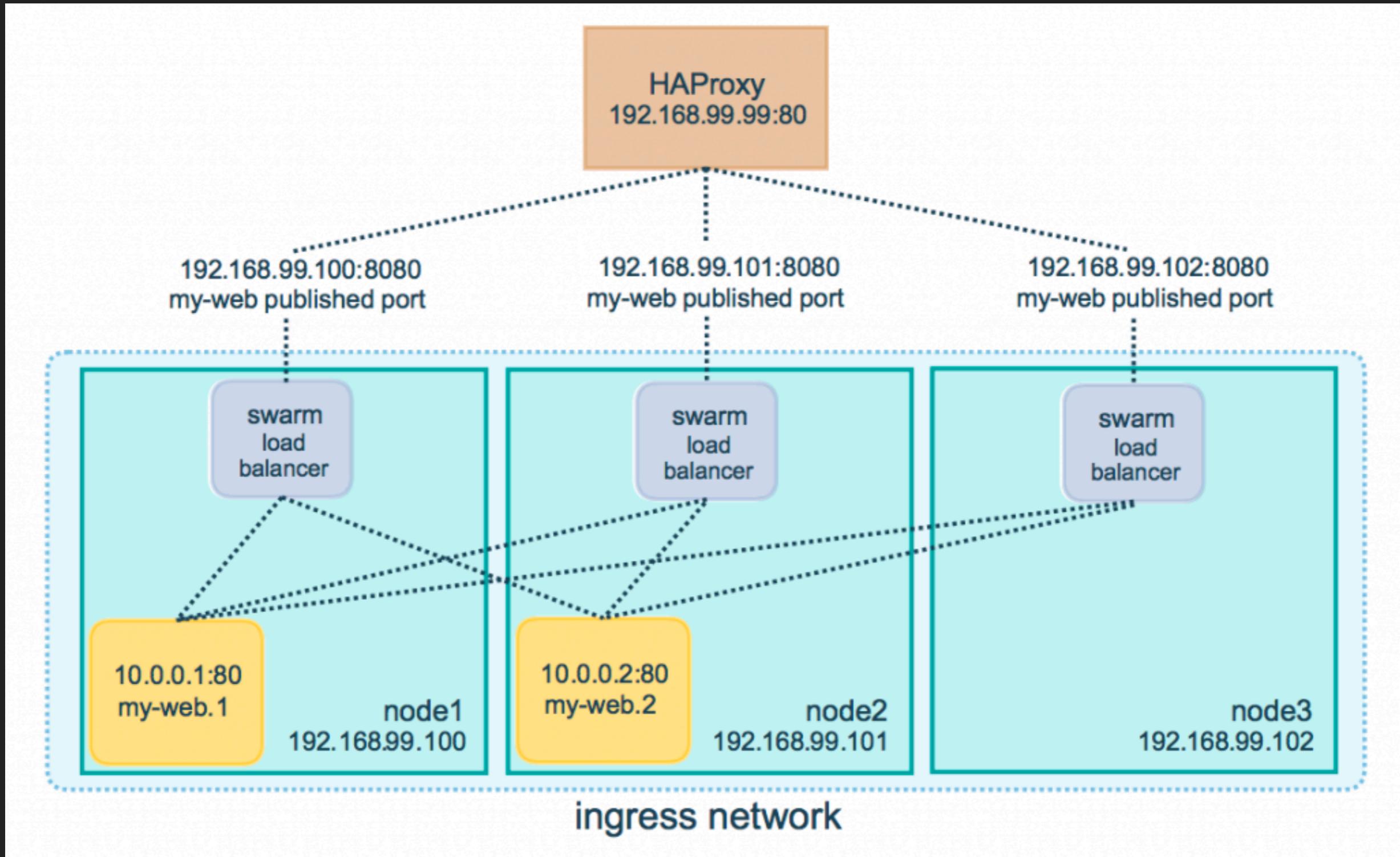
```
[root@alidev ~]# ls -lth /home/dockerstore/volumes/dptest_app-img-data/_data
total 0
[root@alidev ~]# █
```

```
[root@tag01 ~]# ls -lth /home/dockerstore/volumes/dptest_app-img-data/_data
total 20K
-rw-r--r--. 1 root root 17K Sep  9 16:24 u.png
[root@tag01 ~]# █
```

DOCKER SWARM



DOCKER SWARM



DOCKER ENGINE PLUGINS

► https://docs.docker.com/engine/extend/legacy_plugins/

Network plugins

Plugin	Description
Contiv Networking	An open source network plugin to provide infrastructure and security policies for a multi-tenant micro services deployment, while providing an integration to physical network for non-container workload. Contiv Networking implements the remote driver and IPAM APIs available in Docker 1.9 onwards.
Kuryr Network Plugin	A network plugin is developed as part of the OpenStack Kuryr project and implements the Docker networking (libnetwork) remote driver API by utilizing Neutron, the OpenStack networking service. It includes an IPAM driver as well.
Weave Network Plugin	A network plugin that creates a virtual network that connects your Docker containers - across multiple hosts or clouds and enables automatic discovery of applications. Weave networks are resilient, partition tolerant, secure and work in partially connected networks, and other adverse environments - all configured with delightful simplicity.

DOCKER ENGINE PLUGINS

Volume plugins

Plugin	Description
Azure File Storage plugin	Lets you mount Microsoft Azure File Storage shares to Docker containers as volumes using the SMB 3.0 protocol. Learn more.
BeeGFS Volume Plugin	An open source volume plugin to create persistent volumes in a BeeGFS parallel file system.
Blockbridge plugin	A volume plugin that provides access to an extensible set of container-based persistent storage options. It supports single and multi-host Docker environments with features that include tenant isolation, automated provisioning, encryption, secure deletion, snapshots and QoS.
Contiv Volume Plugin	An open source volume plugin that provides multi-tenant, persistent, distributed storage with intent based consumption. It has support for Ceph and NFS.
Convoy plugin	A volume plugin for a variety of storage back-ends including device mapper and NFS. It's a simple standalone executable written in Go and provides the framework to support vendor-specific extensions such as snapshots, backups and restore.

CONFIGS

- ▶ <https://docs.docker.com/engine/swarm/configs/>
- ▶ <https://docs.docker.com/engine/swarm/configs/#example-rotate-a-config>
- ▶ Issue
 - ▶ <https://blog.sunekeller.dk/2019/01/docker-stack-deploy-update-configs/>
 - ▶ <https://github.com/moby/moby/issues/35048>

NETWORK

▶ Attach a network

```
[xcwang@tag01 ~]$ docker run -it --rm --entrypoint sh --network dptest_app_network raese  
ne/alpine-nettools
```

NETWORK

▶ Attach a network

```
[xcwang@alidev docker-swarm]$ docker stack deploy -c docker-compose-nettools.yaml tool
Creating service tool_nettools
```

! docker-compose-nettools.yaml ×

```
docker-swarm > ! docker-compose-nettools.yaml > ...
1   version: "3.7"
2
3   services:
4     nettools:
5       image: raesene/alpine-nettools
6       networks:
7         - app_network
8
9   networks:
10    app_network:
11      name: dptest_app_network
12      external: true
13
```

```
[xcwang@alidev docker-swarm]$ docker stack ps tool
ID          NAME          IMAGE          NODE
DESIRED STATE  CURRENT STATE  ERROR          PORTS
wcxtsbcqj5k2  tool_nettools.1  raesene/alpine-nettools:latest  alidev
        Running          Preparing 19 seconds ago
[xcwang@alidev docker-swarm]$ docker ps
CONTAINER ID  IMAGE          COMMAND          CREATED
STATUS          PORTS          NAMES
ea825b856b07  raesene/alpine-nettools:latest  "/entrypoint.sh"  7 seconds ago
          o          Up 5 seconds
bb
95a7d53256e1  dp_webapp:latest  "node app.js"  2 hours ago
          bb          Up 2 hours (healthy)  80/tcp
          14
14
99352bdfb713  mysql:5.6      "docker-entrypoint.s..."  2 hours ago
          14          Up 2 hours (healthy)  3306/tcp
          l
```

NETWORK

► DNS

```
[xcwang@alidev docker-swarm]$ docker exec -it ea825b856b07 sh  
/ # dig webapp  
  
; <>> DiG 9.10.8-P1 <>> webapp  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22330  
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0  
  
;; QUESTION SECTION:  
;webapp.                      IN      A  
  
;; ANSWER SECTION:  
webapp.          600      IN      A      10.0.5.5  
  
;; Query time: 0 msec  
;; SERVER: 127.0.0.1#53(127.0.0.11)  
;; WHEN: Mon Sep 09 09:54:09 UTC 2019  
;; MSG SIZE  rcvd: 46
```

NETWORK

► DNS

```
/ # dig tasks.webapp

; <>> DiG 9.10.8-P1 <>> tasks.webapp
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62662
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;tasks.webapp.           IN      A

;; ANSWER SECTION:
tasks.webapp.        600    IN      A      10.0.5.7
tasks.webapp.        600    IN      A      10.0.5.6
tasks.webapp.        600    IN      A      10.0.5.8

;; Query time: 0 msec
;; SERVER: 127.0.0.11#53(127.0.0.11)
;; WHEN: Mon Sep 09 09:54:16 UTC 2019
;; MSG SIZE  rcvd: 114

/ # █
```