

OFFICIAL REPOSITORY

swarm (/r/ /swarm/) ☆

Last pushed: 4 months ago

Repo Info (/ /swarm/)

Short Description

Swarm: a Docker-native clustering system.

Full Description

Supported tags and respective Dockerfile links

- 1.2.9 latest (*Dockerfile*) (<https://github.com/docker/swarm-library-image/blob/a2a08d733833327354b5d3336ffbd19e4f369a2d/Dockerfile>)

Quick reference

- **Where to get help:**
[the Docker Community Forums \(https://forums.docker.com/\)](https://forums.docker.com/), [the Docker Community Slack \(https://blog.docker.com/2016/11/introducing-docker-community-directory-docker-community-slack/\)](https://blog.docker.com/2016/11/introducing-docker-community-directory-docker-community-slack/), or [Stack Overflow \(https://stackoverflow.com/search?tab=newest&q=docker\)](https://stackoverflow.com/search?tab=newest&q=docker)
- **Where to file issues:**
<https://github.com/docker/swarm-library-image/issues> (<https://github.com/docker/swarm-library-image/issues>)
- **Maintained by:**
[Docker, Inc. \(https://github.com/docker/swarm-library-image\)](https://github.com/docker/swarm-library-image)
- **Supported architectures:** ([more info \(https://github.com/docker-library/official-images#architectures-other-than-amd64\)](https://github.com/docker-library/official-images#architectures-other-than-amd64))
[amd64 \(https://hub.docker.com/r/amd64/swarm/\)](https://hub.docker.com/r/amd64/swarm/)

- **Published image artifact details:**

[repo-info repo's repos/swarm/ directory](https://github.com/docker-library/repo-info/blob/master/repos/swarm) (<https://github.com/docker-library/repo-info/blob/master/repos/swarm>) ([history](https://github.com/docker-library/repo-info/commits/master/repos/swarm) (<https://github.com/docker-library/repo-info/commits/master/repos/swarm>))
(image metadata, transfer size, etc)

- **Image updates:**

[official-images PRs with label library/swarm](https://github.com/docker-library/official-images/pulls?q=label%3Alibrary%2Fswarm) (<https://github.com/docker-library/official-images/pulls?q=label%3Alibrary%2Fswarm>)
[official-images repo's library/swarm file](https://github.com/docker-library/official-images/blob/master/library/swarm) (<https://github.com/docker-library/official-images/blob/master/library/swarm>) ([history](https://github.com/docker-library/official-images/commits/master/library/swarm) (<https://github.com/docker-library/official-images/commits/master/library/swarm>))

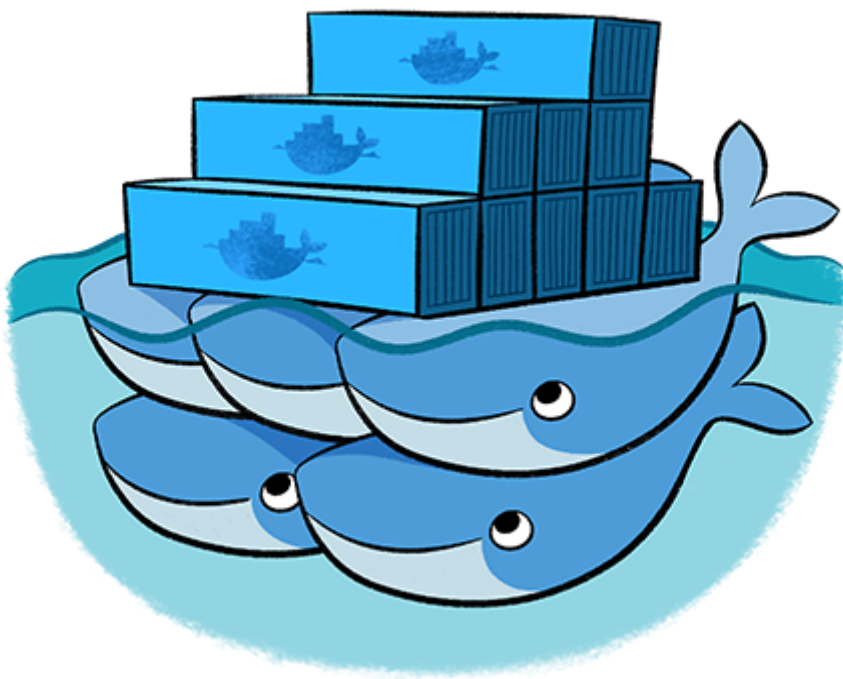
- **Source of this description:**

[docs repo's swarm/ directory](https://github.com/docker-library/docs/tree/master/swarm) (<https://github.com/docker-library/docs/tree/master/swarm>) ([history](https://github.com/docker-library/docs/commits/master/swarm) (<https://github.com/docker-library/docs/commits/master/swarm>))

- **Supported Docker versions:**

[the latest release](https://github.com/docker/docker-ce/releases/latest) (<https://github.com/docker/docker-ce/releases/latest>) (down to 1.6 on a best-effort basis)

Swarm: a Docker-native clustering system



swarm is a simple tool which controls a cluster of Docker hosts and exposes it as a single "virtual" host.

swarm uses the standard Docker API as its frontend, which means any tool which speaks Docker can control swarm transparently: dokku, docker-compose, krane, flynn, deis, docker-ui, shipyard, drone.io, Jenkins... and of course the Docker client itself.

Like the other Docker projects, swarm follows the "batteries included but removable" principle. It ships with a simple scheduling backend out of the box, and as initial development settles, an API will develop to enable pluggable backends. The goal is to provide a smooth out-of-box experience for simple use cases, and allow swapping in more powerful backends, like Mesos, for large scale production deployments.

Example usage

```
# create a cluster
$ docker run --rm swarm create
6856663cdefdec325839a4b7e1de38e8 # <- this is your unique <cluster_id>

# on each of your nodes, start the swarm agent
# <node_ip> doesn't have to be public (eg. 192.168.0.X),
# as long as the swarm manager can access it.
$ docker run -d swarm join --addr=<node_ip:2375> token://<cluster_id>

# start the manager on any machine or your laptop
$ docker run -t -p <swarm_port>:2375 -t swarm manage token://<cluster_id>

# use the regular docker cli
$ docker -H tcp://<swarm_ip:swarm_port> info
$ docker -H tcp://<swarm_ip:swarm_port> run ...
$ docker -H tcp://<swarm_ip:swarm_port> ps
$ docker -H tcp://<swarm_ip:swarm_port> logs ...
...

# list nodes in your cluster
$ docker run --rm swarm list token://<cluster_id>
<node_ip:2375>
```

See [here \(https://github.com/docker/swarm/blob/master/discovery/README.md\)](https://github.com/docker/swarm/blob/master/discovery/README.md) for more information about other discovery services.

Advanced Scheduling

See [filters \(https://github.com/docker/swarm/blob/master/scheduler/filter/README.md\)](https://github.com/docker/swarm/blob/master/scheduler/filter/README.md) and [strategies \(https://github.com/docker/swarm/blob/master/scheduler/strategy/README.md\)](https://github.com/docker/swarm/blob/master/scheduler/strategy/README.md) to learn more about advanced scheduling.

TLS

Swarm supports TLS authentication between the CLI and Swarm but also between Swarm and the Docker nodes.

In order to enable TLS, the same command line options as Docker can be specified:

```
swarm manage --tlsverify --tlscacert=<CACERT> --tlscert=<CERT> --  
tlskey=<KEY> [...]
```

Please refer to the [Docker documentation \(https://docs.docker.com/articles/https/\)](https://docs.docker.com/articles/https/) for more information on how to set up TLS authentication on Docker and generating the certificates.

Note that Swarm certificates must be generated with `extendedKeyUsage = clientAuth, serverAuth`.

License

View [license information \(https://github.com/docker/swarm/blob/master/LICENSE.code\)](https://github.com/docker/swarm/blob/master/LICENSE.code) for the software contained in this image.

As with all Docker images, these likely also contain other software which may be under other licenses (such as Bash, etc from the base distribution, along with any direct or indirect dependencies of the primary software being contained).

Some additional license information which was able to be auto-detected might be found in the [repo-info repository's swarm/ directory \(https://github.com/docker-library/repo-info/tree/master/repos/swarm\)](https://github.com/docker-library/repo-info/tree/master/repos/swarm).

As for any pre-built image usage, it is the image user's responsibility to ensure that any use of this image complies with any relevant licenses for all software contained within.

Docker Pull Command



```
docker pull swarm
```