

# docker swarm init

*Estimated reading time: 7 minutes*

## Description

Initialize a swarm

**API 1.24+** (<https://docs.docker.com/engine/api/v1.24/>) The client and daemon API must both be at least 1.24 (<https://docs.docker.com/engine/api/v1.24/>) to use this command. Use the `docker version` command on the client to check your client and daemon API versions.

**Swarm** This command works with the Swarm orchestrator.

## Usage

```
docker swarm init [OPTIONS]
```

## Options

Name, shorthand	Default	Description
<code>--advertise-addr</code>		Advertised address (format: <ip interface>[:port])
<code>--autolock</code>		Enable manager autolocking (requiring an unlock key to start a stopped manager)
<code>--availability</code>	<code>active</code>	Availability of the node ("active" "pause" "drain")
<code>--cert-expiry</code>	<code>2160h0m0s</code>	Validity period for node certificates (ns us ms s m h)
<code>--data-path-addr</code>		Address or interface to use for data path traffic (format: <ip interface>)
<code>--dispatcher-heartbeat</code>	<code>5s</code>	Dispatcher heartbeat period (ns us ms s m h)

Name, shorthand	Default	Description
<code>--external-ca</code>		Specifications of one or more certificate signing endpoints
<code>--force-new-cluster</code>		Force create a new cluster from current state
<code>--listen-addr</code>	<code>0.0.0.0:2377</code>	Listen address (format: <ip   interface> [:port])
<code>--max-snapshots</code>		API 1.25+ ( <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> ) Number of additional Raft snapshots to retain
<code>--snapshot-interval</code>	<code>10000</code>	API 1.25+ ( <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> ) Number of log entries between Raft snapshots
<code>--task-history-limit</code>	<code>5</code>	Task history retention limit

## Parent command

Command	Description
<code>docker swarm</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm/">https://docs.docker.com/engine/reference/commandline/swarm/</a> )	Manage Swarm

## Related commands

Command	Description
<code>docker swarm ca</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_ca/">https://docs.docker.com/engine/reference/commandline/swarm_ca/</a> )	Display and rotate the root CA
<code>docker swarm init</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_init/">https://docs.docker.com/engine/reference/commandline/swarm_init/</a> )	Initialize a swarm
<code>docker swarm join</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_join/">https://docs.docker.com/engine/reference/commandline/swarm_join/</a> )	Join a swarm as a node and/or manager

Command	Description
<code>docker swarm join-token</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_join-token/">https://docs.docker.com/engine/reference/commandline/swarm_join-token/</a> )	Manage join tokens
<code>docker swarm leave</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_leave/">https://docs.docker.com/engine/reference/commandline/swarm_leave/</a> )	Leave the swarm
<code>docker swarm unlock</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_unlock/">https://docs.docker.com/engine/reference/commandline/swarm_unlock/</a> )	Unlock swarm
<code>docker swarm unlock-key</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_unlock-key/">https://docs.docker.com/engine/reference/commandline/swarm_unlock-key/</a> )	Manage the unlock key
<code>docker swarm update</code> ( <a href="https://docs.docker.com/engine/reference/commandline/swarm_update/">https://docs.docker.com/engine/reference/commandline/swarm_update/</a> )	Update the swarm

## Extended description

Initialize a swarm. The docker engine targeted by this command becomes a manager in the newly created single-node swarm.

## Examples

```
$ docker swarm init --advertise-addr 192.168.99.121
Swarm initialized: current node (bvz81updecsj6wjz393c09vti) is now a manager.
```

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

```
docker swarm join \
  --token SWMTKN-1-3pu6hszjas19xyp7ghgosyx9k8atbfcr8p2is99znp26u2lk1-1awxwu
  172.17.0.2:2377
```

To add a manager to this swarm, run 'docker swarm join-token manager' and foll



`docker swarm init` generates two random tokens, a worker token and a manager token. When you join a new node to the swarm, the node joins as a worker or manager node based upon the token you pass to `swarm join` ([https://docs.docker.com/engine/reference/commandline/swarm\\_join/](https://docs.docker.com/engine/reference/commandline/swarm_join/)).

After you create the swarm, you can display or rotate the token using `swarm join-token` ([https://docs.docker.com/engine/reference/commandline/swarm\\_join\\_token/](https://docs.docker.com/engine/reference/commandline/swarm_join_token/)).

### --autolock

This flag enables automatic locking of managers with an encryption key. The private keys and data stored by all managers will be protected by the encryption key printed in the output, and will not be accessible without it. Thus, it is very important to store this key in order to activate a manager after it restarts. The key can be passed to `docker swarm unlock` to reactivate the manager. Autolock can be disabled by running `docker swarm update --autolock=false`. After disabling it, the encryption key is no longer required to start the manager, and it will start up on its own without user intervention.

## `--cert-expiry`

This flag sets the validity period for node certificates.

## `--dispatcher-heartbeat`

This flag sets the frequency with which nodes are told to use as a period to report their health.

## `--external-ca`

This flag sets up the swarm to use an external CA to issue node certificates. The value takes the form `protocol=X,url=Y`. The value for `protocol` specifies what protocol should be used to send signing requests to the external CA. Currently, the only supported value is `cfssl`. The URL specifies the endpoint where signing requests should be submitted.

## `--force-new-cluster`

This flag forces an existing node that was part of a quorum that was lost to restart as a single node Manager without losing its data.

## `--listen-addr`

The node listens for inbound swarm manager traffic on this address. The default is to listen on 0.0.0.0:2377. It is also possible to specify a network interface to listen on that interface's address; for example `--listen-addr eth0:2377`.

Specifying a port is optional. If the value is a bare IP address or interface name, the default port 2377 will be used.

## `--advertise-addr`

This flag specifies the address that will be advertised to other members of the swarm for API access and overlay networking. If unspecified, Docker will check if the system has a single IP address, and use that IP address with the listening port (see `--listen-addr`). If the system has multiple IP addresses, `--advertise-addr` must be specified so that the correct address is chosen for inter-manager communication and overlay networking.

It is also possible to specify a network interface to advertise that interface's address; for example `--advertise-addr eth0:2377` .

Specifying a port is optional. If the value is a bare IP address or interface name, the default port 2377 will be used.

### `--data-path-addr`

This flag specifies the address that global scope network drivers will publish towards other nodes in order to reach the containers running on this node. Using this parameter it is then possible to separate the container's data traffic from the management traffic of the cluster. If unspecified, Docker will use the same IP address or interface that is used for the advertise address.

### `--task-history-limit`

This flag sets up task history retention limit.

### `--max-snapshots`

This flag sets the number of old Raft snapshots to retain in addition to the current Raft snapshots. By default, no old snapshots are retained. This option may be used for debugging, or to store old snapshots of the swarm state for disaster recovery purposes.

### `--snapshot-interval`

This flag specifies how many log entries to allow in between Raft snapshots. Setting this to a higher number will trigger snapshots less frequently. Snapshots compact the Raft log and allow for more efficient transfer of the state to new managers. However, there is a performance cost to taking snapshots frequently.

### `--availability`

This flag specifies the availability of the node at the time the node joins a master.

Possible availability values are `active` , `pause` , or `drain` .

This flag is useful in certain situations. For example, a cluster may want to have dedicated manager nodes that are not served as worker nodes. This could be achieved by passing `--availability=drain` to `docker swarm init` .