Docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:

```
      --config string           Location of client config files (default "/home/sk/.docker")
 -D, --debug                    Enable debug mode
 -H, --host list                Daemon socket(s) to connect to
 -l, --log-level string         Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
      --tls                     Use TLS; implied by --tlsverify
      --tlscacert string        Trust certs signed only by this CA (default "/home/sk/.docker/ca.pem")
      --tlscert string          Path to TLS certificate file (default "/home/sk/.docker/cert.pem")
      --tlskey string           Path to TLS key file (default "/home/sk/.docker/key.pem")
      --tlsverify               Use TLS and verify the remote
 -v, --version                  Print version information and quit
```

Management Commands:

```
  config      Manage Docker configs
  container   Manage containers
  image       Manage images
  network     Manage networks
  node        Manage Swarm nodes
  plugin      Manage plugins
  secret      Manage Docker secrets
  service     Manage services
  stack       Manage Docker stacks
  swarm       Manage Swarm
  system      Manage Docker
  trust       Manage trust on Docker images
  volume      Manage volumes
```

Commands:

```
  attach      Attach local standard input, output, and error streams to a running container
  build       Build an image from a Dockerfile
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
  images      List images
```

import	Import the contents from a tarball to create a filesystem image
info	Display system-wide information
inspect	Return low-level information on Docker objects
kill	Kill one or more running containers
load	Load an image from a tar archive or STDIN
login	Log in to a Docker registry
logout	Log out from a Docker registry
logs	Fetch the logs of a container
pause	Pause all processes within one or more containers
port	List port mappings or a specific mapping for the container
ps	List containers
pull	Pull an image or a repository from a registry
push	Push an image or a repository to a registry
rename	Rename a container
restart	Restart one or more containers
rm	Remove one or more containers
rmi	Remove one or more images
run	Run a command in a new container
save	Save one or more images to a tar archive (streamed to STDOUT by default)
search	Search the Docker Hub for images
start	Start one or more stopped containers
stats	Display a live stream of container(s) resource usage statistics
stop	Stop one or more running containers
tag	Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
top	Display the running processes of a container
unpause	Unpause all processes within one or more containers
update	Update configuration of one or more containers
version	Show the Docker version information
wait	Block until one or more containers stop, then print their exit codes

Docker image COMMAND

Manage images

Commands:

build	Build an image from a Dockerfile
history	Show the history of an image
import	Import the contents from a tarball to create a filesystem image
inspect	Display detailed information on one or more images
load	Load an image from a tar archive or STDIN
ls	List images

```
  prune       Remove unused images
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rm          Remove one or more images
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
```

Docker build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile

Options:

```
        --add-host list           Add a custom host-to-IP mapping (host:ip)
        --build-arg list          Set build-time variables
        --cache-from strings      Images to consider as cache sources
        --cgroup-parent string Optional parent cgroup for the container
        --compress                Compress the build context using gzip
        --cpu-period int          Limit the CPU CFS (Completely Fair Scheduler) period
        --cpu-quota int           Limit the CPU CFS (Completely Fair Scheduler) quota
  -c, --cpu-shares int            CPU shares (relative weight)
        --cpuset-cpus string      CPUs in which to allow execution (0-3, 0,1)
        --cpuset-mems string            MEMs in which to allow execution (0-3, 0,1)
        --disable-content-trust   Skip image verification (default true)
  -f, --file string           Name of the Dockerfile (Default is 'PATH/Dockerfile')
        --force-rm                Always remove intermediate containers
        --iidfile string          Write the image ID to the file
        --isolation string        Container isolation technology
        --label list              Set metadata for an image
  -m, --memory bytes              Memory limit
        --memory-swap bytes             Swap limit equal to memory plus swap: '-1' to enable
unlimited swap
        --network string          Set the networking mode for the RUN instructions during build
(default "default")
        --no-cache                Do not use cache when building the image
        --pull                    Always attempt to pull a newer version of the image
  -q, --quiet                 Suppress the build output and print image ID on success
        --rm                            Remove intermediate containers after a successful build (default
true)
        --security-opt strings    Security options
        --shm-size bytes          Size of /dev/shm
  -t, --tag list              Name and optionally a tag in the 'name:tag' format
```

```
      --target string          Set the target build stage to build.
      --ulimit ulimit          Ulimit options (default [])
```

Docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Run a command in a new container

Options:

```
      --add-host list                Add a custom host-to-IP mapping (host:ip)
  -a, --attach list                  Attach to STDIN, STDOUT or STDERR
      --blkio-weight uint16          Block IO (relative weight), between 10 and 1000, or 0 to disable
(default 0)
      --blkio-weight-device list     Block IO weight (relative device weight) (default [])
      --cap-add list                 Add Linux capabilities
      --cap-drop list                Drop Linux capabilities
      --cgroup-parent string         Optional parent cgroup for the container
      --cidfile string               Write the container ID to the file
      --cpu-period int               Limit CPU CFS (Completely Fair Scheduler) period
      --cpu-quota int                Limit CPU CFS (Completely Fair Scheduler) quota
      --cpu-rt-period int            Limit CPU real-time period in microseconds
      --cpu-rt-runtime int           Limit CPU real-time runtime in microseconds
  -c, --cpu-shares int               CPU shares (relative weight)
      --cpus decimal                 Number of CPUs
      --cpuset-cpus string           CPUs in which to allow execution (0-3, 0,1)
      --cpuset-mems string           MEMs in which to allow execution (0-3, 0,1)
  -d, --detach                       Run container in background and print container ID
      --detach-keys string           Override the key sequence for detaching a container
      --device list                  Add a host device to the container
      --device-cgroup-rule list      Add a rule to the cgroup allowed devices list
      --device-read-bps list         Limit read rate (bytes per second) from a device (default [])
      --device-read-iops list        Limit read rate (IO per second) from a device (default [])
      --device-write-bps list        Limit write rate (bytes per second) to a device (default [])
      --device-write-iops list       Limit write rate (IO per second) to a device (default [])
      --disable-content-trust        Skip image verification (default true)
      --dns list                     Set custom DNS servers
      --dns-option list              Set DNS options
      --dns-search list              Set custom DNS search domains
      --entrypoint string            Overwrite the default ENTRYPOINT of the image
  -e, --env list                     Set environment variables
      --env-file list                Read in a file of environment variables
      --expose list                  Expose a port or a range of ports
      --group-add list               Add additional groups to join
```

```
      --health-cmd string              Command to run to check health
      --health-interval duration       Time between running the check (ms|s|m|h) (default 0s)
      --health-retries int             Consecutive failures needed to report unhealthy
      --health-start-period duration   Start period for the container to initialize before starting
health-retries countdown (ms|s|m|h) (default 0s)
      --health-timeout duration        Maximum time to allow one check to run (ms|s|m|h)
(default 0s)
      --help                           Print usage
  -h, --hostname string                Container host name
      --init                           Run an init inside the container that forwards signals and reaps
processes
  -i, --interactive                    Keep STDIN open even if not attached
      --ip string                      IPv4 address (e.g., 172.30.100.104)
      --ip6 string                     IPv6 address (e.g., 2001:db8::33)
      --ipc string                     IPC mode to use
      --isolation string               Container isolation technology
      --kernel-memory bytes            Kernel memory limit
  -l, --label list                     Set meta data on a container
      --label-file list                Read in a line delimited file of labels
      --link list                      Add link to another container
      --link-local-ip list             Container IPv4/IPv6 link-local addresses
      --log-driver string              Logging driver for the container
      --log-opt list                   Log driver options
      --mac-address string             Container MAC address (e.g., 92:d0:c6:0a:29:33)
  -m, --memory bytes                   Memory limit
      --memory-reservation bytes       Memory soft limit
      --memory-swap bytes              Swap limit equal to memory plus swap: '-1' to enable
unlimited swap
      --memory-swappiness int          Tune container memory swappiness (0 to 100) (default -1)
      --mount mount                    Attach a filesystem mount to the container
      --name string                    Assign a name to the container
      --network string                 Connect a container to a network (default "default")
      --network-alias list             Add network-scoped alias for the container
      --no-healthcheck                 Disable any container-specified HEALTHCHECK
      --oom-kill-disable               Disable OOM Killer
      --oom-score-adj int              Tune host's OOM preferences (-1000 to 1000)
      --pid string                     PID namespace to use
      --pids-limit int                 Tune container pids limit (set -1 for unlimited)
      --privileged                     Give extended privileges to this container
  -p, --publish list                   Publish a container's port(s) to the host
  -P, --publish-all                    Publish all exposed ports to random ports
      --read-only                      Mount the container's root filesystem as read only
      --restart string                 Restart policy to apply when a container exits (default "no")
```

--rm	Automatically remove the container when it exits
--runtime string	Runtime to use for this container
--security-opt list	Security Options
--shm-size bytes	Size of /dev/shm
--sig-proxy	Proxy received signals to the process (default true)
--stop-signal string	Signal to stop a container (default "SIGTERM")
--stop-timeout int	Timeout (in seconds) to stop a container
--storage-opt list	Storage driver options for the container
--sysctl map	Sysctl options (default map[])
--tmpfs list	Mount a tmpfs directory
-t, --tty	Allocate a pseudo-TTY
--ulimit ulimit	Ulimit options (default [])
-u, --user string	Username or UID (format: <name|uid>[:<group|gid>])
--userns string	User namespace to use
--uts string	UTS namespace to use
-v, --volume list	Bind mount a volume
--volume-driver string	Optional volume driver for the container
--volumes-from list	Mount volumes from the specified container(s)
-w, --workdir string	Working directory inside the container

Docker container COMMAND

docker container --help
Manage containers

Commands:

attach	Attach local standard input, output, and error streams to a running container
commit	Create a new image from a container's changes
cp	Copy files/folders between a container and the local filesystem
create	Create a new container
diff	Inspect changes to files or directories on a container's filesystem
exec	Run a command in a running container
export	Export a container's filesystem as a tar archive
inspect	Display detailed information on one or more containers
kill	Kill one or more running containers
logs	Fetch the logs of a container
ls	List containers
pause	Pause all processes within one or more containers
port	List port mappings or a specific mapping for the container
prune	Remove all stopped containers
rename	Rename a container
restart	Restart one or more containers
rm	Remove one or more containers

run	Run a command in a new container
start	Start one or more stopped containers
stats	Display a live stream of container(s) resource usage statistics
stop	Stop one or more running containers
top	Display the running processes of a container
unpause	Unpause all processes within one or more containers
update	Update configuration of one or more containers
wait	Block until one or more containers stop, then print their exit codes

Docker service COMMAND

Manage services

Commands:

create	Create a new service
inspect	Display detailed information on one or more services
logs	Fetch the logs of a service or task
ls	List services
ps	List the tasks of one or more services
rm	Remove one or more services
rollback	Revert changes to a service's configuration
scale	Scale one or multiple replicated services
update	Update a service

Docker volume COMMAND

Manage volumes

Commands:

create	Create a volume
inspect	Display detailed information on one or more volumes
ls	List volumes
prune	Remove all unused local volumes
rm	Remove one or more volumes

Docker-compose

Define and run multi-container applications with Docker.

Usage:

```
docker-compose [-f <arg>...] [options] [COMMAND] [ARGS...]
docker-compose -h|--help
```

Options:

```
-f, --file FILE             Specify an alternate compose file (default: docker-compose.yml)
-p, --project-name NAME     Specify an alternate project name (default: directory name)
--verbose                   Show more output
--no-ansi                   Do not print ANSI control characters
-v, --version               Print version and exit
-H, --host HOST                 Daemon socket to connect to

--tls                       Use TLS; implied by --tlsverify
--tlscacert CA_PATH         Trust certs signed only by this CA
--tlscert CLIENT_CERT_PATH  Path to TLS certificate file
--tlskey TLS_KEY_PATH       Path to TLS key file
--tlsverify                 Use TLS and verify the remote
--skip-hostname-check       Don't check the daemon's hostname against the name specified
                            in the client certificate (for example if your docker host
                            is an IP address)
--project-directory PATH    Specify an alternate working directory
                            (default: the path of the Compose file)
```

Commands:

```
build              Build or rebuild services
bundle             Generate a Docker bundle from the Compose file
config             Validate and view the Compose file
create             Create services
down               Stop and remove containers, networks, images, and volumes
events             Receive real time events from containers
exec               Execute a command in a running container
help               Get help on a command
images             List images
kill               Kill containers
logs               View output from containers
pause              Pause services
port               Print the public port for a port binding
ps                 List containers
pull               Pull service images
push               Push service images
restart            Restart services
rm                 Remove stopped containers
```

run	Run a one-off command
scale	Set number of containers for a service
start	Start services
stop	Stop services
top	Display the running processes
unpause	Unpause services
up	Create and start containers
version	Show the Docker-Compose version information

Docker swarm COMMAND

Commands:

ca	Display and rotate the root CA
init	Initialize a swarm
join	Join a swarm as a node and/or manager
join-token	Manage join tokens
leave	Leave the swarm
unlock	Unlock swarm
unlock-key	Manage the unlock key
update	Update the swarm

Docker stack COMMAND

Manage Docker stacks

Options:

--orchestrator string Orchestrator to use (swarm|kubernetes|all)

Commands:

deploy	Deploy a new stack or update an existing stack
ls	List stacks
ps	List the tasks in the stack
rm	Remove one or more stacks
services	List the services in the stack

Docker network COMMAND

Manage networks

Commands:

connect	Connect a container to a network
create	Create a network
disconnect	Disconnect a container from a network
inspect	Display detailed information on one or more networks
ls	List networks
prune	Remove all unused networks
rm	Remove one or more networks