Saturday, June 6, 2020

Docker

Creating and Using Containers

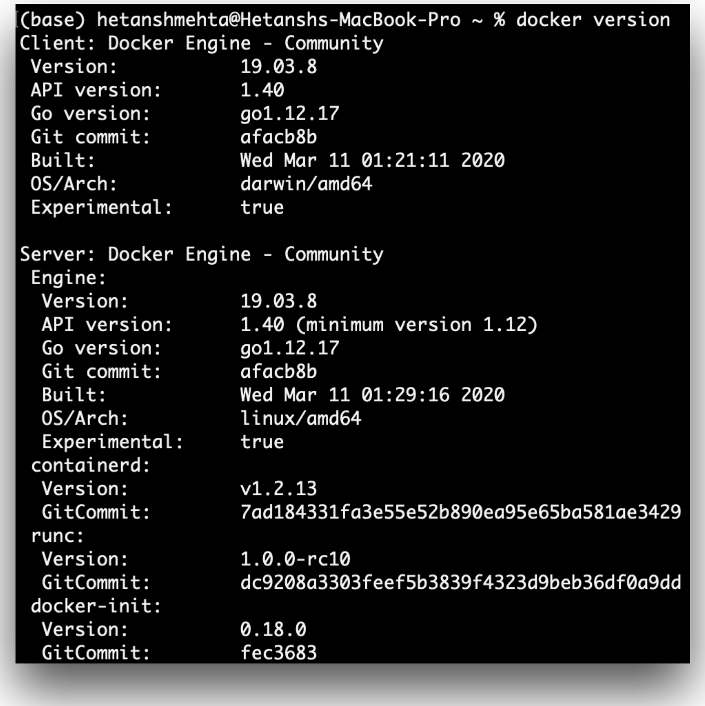
- Check versions of our docker cli and engine
- Create a Nginx (web server) container
- Common container management commands
- Docker Networking Basics

Docker Commands - Basic

1. **docker version**

   This command simply returns the version of your client and the server (also called the engine) which runs in the background on your machine - called a daemon on macOS.

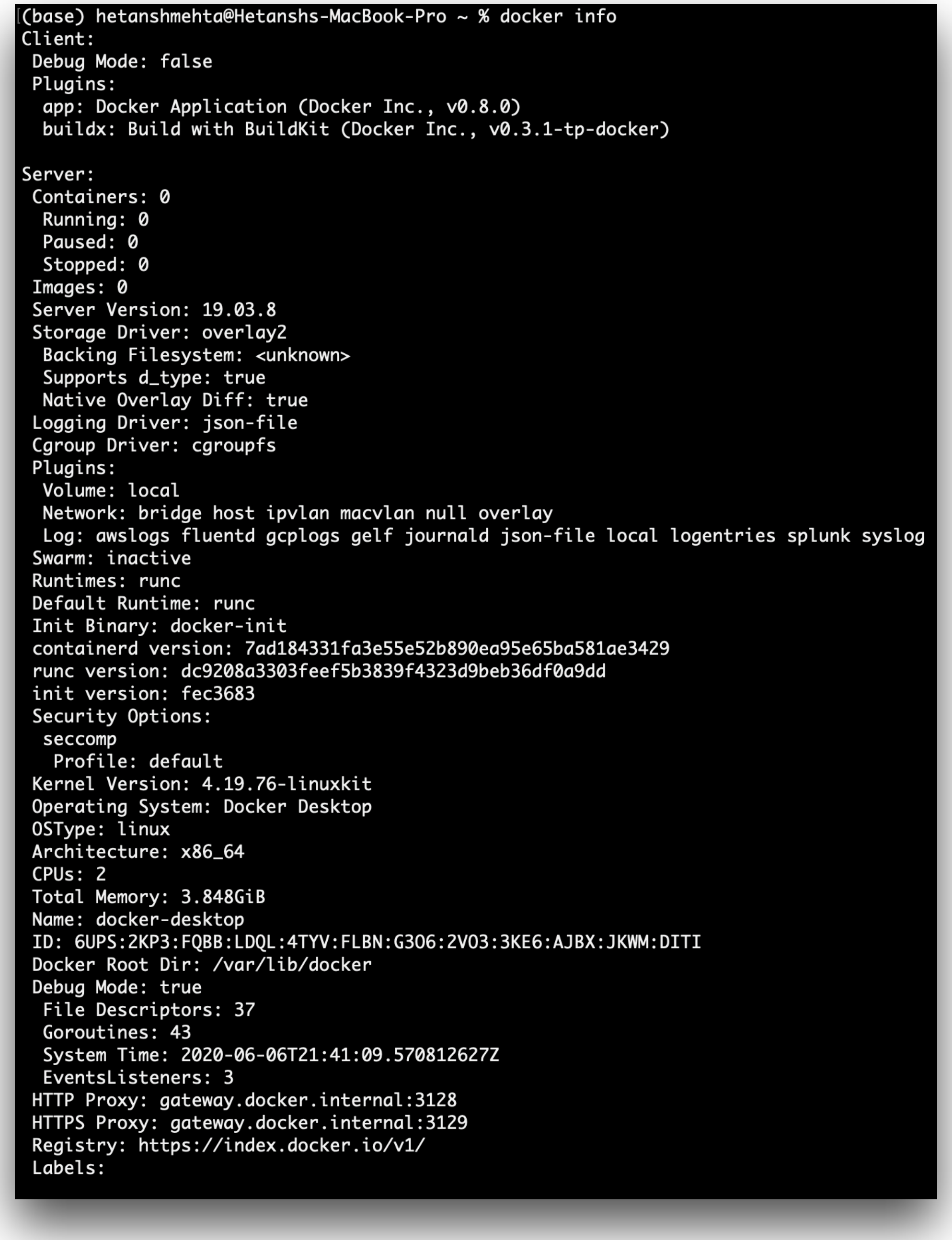
   Our docker command line is actually talking to the server on the machine and returning its values as well as the client's values as well.



```
(base) hetanshmehta@Hetanshs-MacBook-Pro ~ % docker version
Client: Docker Engine - Community
 Version:           19.03.8
 API version:       1.40
 Go version:        go1.12.17
 Git commit:        afacb8b
 Built:             Wed Mar 11 01:21:11 2020
 OS/Arch:           darwin/amd64
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          19.03.8
  API version:      1.40 (minimum version 1.12)
  Go version:       go1.12.17
  Git commit:       afacb8b
  Built:            Wed Mar 11 01:29:16 2020
  OS/Arch:          linux/amd64
  Experimental:     true
 containerd:
  Version:          v1.2.13
  GitCommit:        7ad184331fa3e55e52b890ea95e65ba581ae3429
 runc:
  Version:          1.0.0-rc10
  GitCommit:        dc9208a3303feef5b3839f4323d9beb36df0a9dd
 docker-init:
  Version:          0.18.0
  GitCommit:        fec3683
```

2. **docker info**
   This command returns a lot more stuff than just the version. It'll actually give us a lot of details about our configuration and setup for our engine.

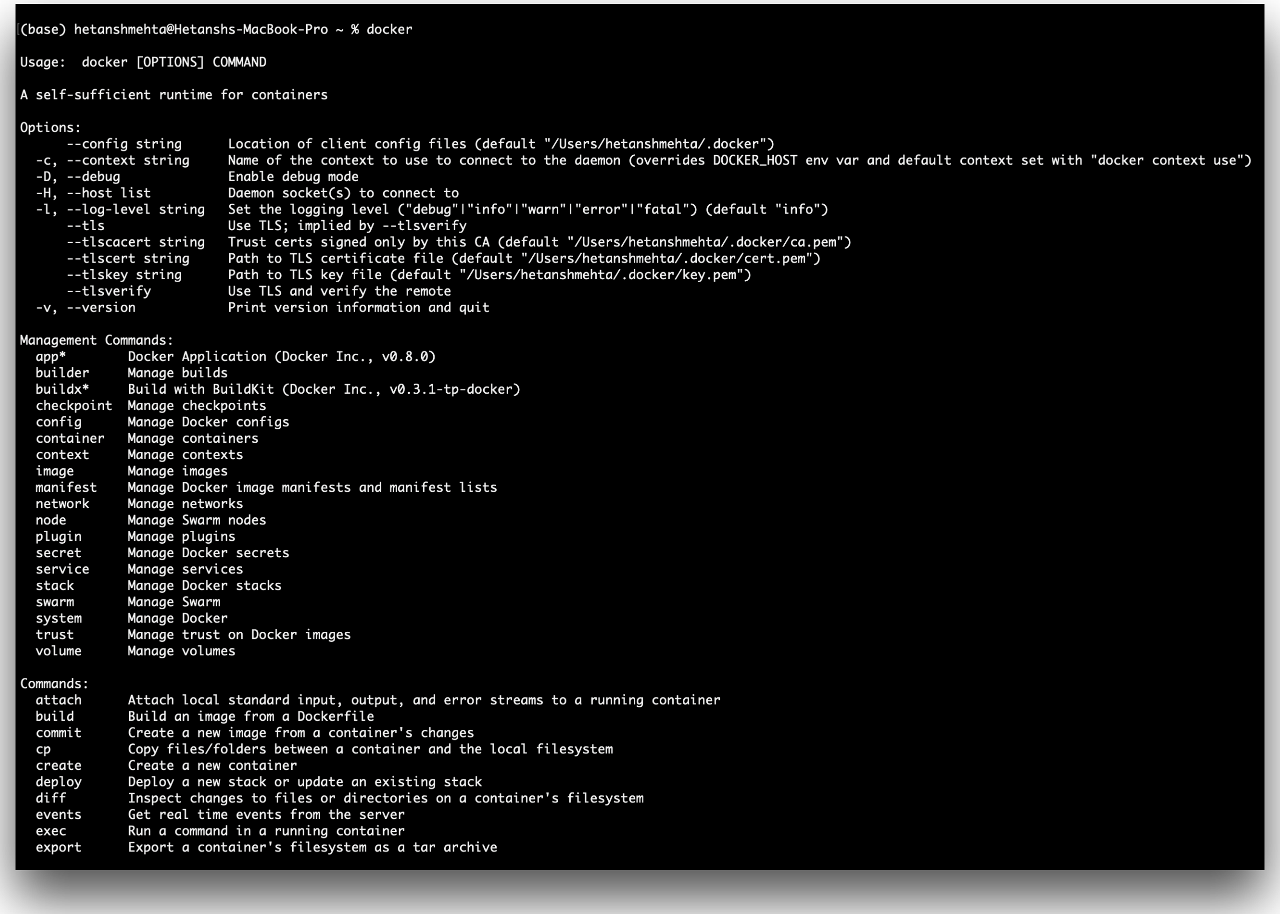


```
[(base) hetanshmehta@Hetanshs-MacBook-Pro ~ % docker info
Client:
 Debug Mode: false
 Plugins:
  app: Docker Application (Docker Inc., v0.8.0)
  buildx: Build with BuildKit (Docker Inc., v0.3.1-tp-docker)

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 19.03.8
 Storage Driver: overlay2
  Backing Filesystem: <unknown>
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: runc
 Default Runtime: runc
 Init Binary: docker-init
 containerd version: 7ad184331fa3e55e52b890ea95e65ba581ae3429
 runc version: dc9208a3303feef5b3839f4323d9beb36df0a9dd
 init version: fec3683
 Security Options:
  seccomp
   Profile: default
 Kernel Version: 4.19.76-linuxkit
 Operating System: Docker Desktop
 OSType: linux
 Architecture: x86_64
 CPUs: 2
 Total Memory: 3.848GiB
 Name: docker-desktop
 ID: 6UPS:2KP3:FQBB:LDQL:4TYV:FLBN:G3O6:2VO3:3KE6:AJBX:JKWM:DITI
 Docker Root Dir: /var/lib/docker
 Debug Mode: true
  File Descriptors: 37
  Goroutines: 43
  System Time: 2020-06-06T21:41:09.570812627Z
  EventsListeners: 3
 HTTP Proxy: gateway.docker.internal:3128
 HTTPS Proxy: gateway.docker.internal:3129
 Registry: https://index.docker.io/v1/
 Labels:
```

3. **`docker`**

   This returns a list of all the commands you can use in Docker.



```
(base) hetanshmehta@Hetanshs-MacBook-Pro ~ % docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string      Location of client config files (default "/Users/hetanshmehta/.docker")
  -c, --context string     Name of the context to use to connect to the daemon (overrides DOCKER_HOST env var and default context set with "docker context use")
  -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 "/Users/hetanshmehta/.docker/ca.pem")
      --tlscert string     Path to TLS certificate file (default "/Users/hetanshmehta/.docker/cert.pem")
      --tlskey string      Path to TLS key file (default "/Users/hetanshmehta/.docker/key.pem")
      --tlsverify          Use TLS and verify the remote
  -v, --version            Print version information and quit

Management Commands:
  app*        Docker Application (Docker Inc., v0.8.0)
  builder     Manage builds
  buildx*     Build with BuildKit (Docker Inc., v0.3.1-tp-docker)
  checkpoint  Manage checkpoints
  config      Manage Docker configs
  container   Manage containers
  context     Manage contexts
  image       Manage images
  manifest    Manage Docker image manifests and manifest lists
  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
  deploy      Deploy a new stack or update an existing stack
  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
```

Syntax for docker management commands:

NEW: `docker <management-command> <sub-command> (options)`

`OLD: docker <command> (options)`

Starting a Nginx Web Server