Penjelasan perintah :

$ docker ps

digunakan untuk melihat semua lis container

$ docker run hello-world

digunakan untuk menjalankan peintah hello-world

$ docker ps -a

perintah untuk me-list semua images

$ docker container run -idt alpine uptime

$ docker ps

perintah ps selain untuk melihat poses, id container, juga dapat digunakan untuk melihat port pada container

$ docker ps -a

perintah ini untuk menampilkan list semua images

$ docker logs 6c30a3374818

$ docker logs 144ed978a896

docker logs digunkan untuk melihat logs yang berjalan secara daemonized

$ docker pull nginx

perintah pull digunakan untuk men-download suatu file

$ docker images

docker images merupakan template dasar untuk docker container. image ini berisi sistem operasi ataupun aplikasi yang sudah selesai. docker images berfungsi untuk menjalankan kontainer

$ docker run -it nginx /bin/bash

perintah run adalah parameter untuk memulai menjalankan sebuah images menjadi container

$ docker run -d -p 8080:08 --name mynginx nginx

> -d tanpa isian parameter, digunakan untuk menjalankan perintah sebagai daemon (background service, tanpa log di terminal)

> -p 8080:80 untuk port forwarding dari port 8080 tujuan ke port 80 asal apliaksi

> --name mynginx untuk memberi nama container dengan nama nginx (biasanya diguakan untuk memudahkan delete containter)

$ docker exec -it mynginx /bin/bash

> docker exec digunakan untuk memulai suatu perintah baru pada container yang sedang berjalan

> -it digunakan untuk menjalankan container pada node interactive di terminal

$ docker ps -l

perintah ps -l digunakan untuk mengecek docker container list

program

[node1] (local) root@192.168.0.23 ~

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

144ed978a896 mongo:latest "docker-entrypoint.s…" About a minute ago Up About a minute 27017/tcp pwd\_mongo.1.vh0zgtajgad7fca3pspk0cvyh

6c30a3374818 mongo-express:latest "tini -- /docker-ent…" About a minute ago Up About a minute 8081/tcp pwd\_mongo-express.1.pecqu5ykf9ztc9cwnj7c4lcaa

[node1] (local) root@192.168.0.23 ~

$ docker run hello-world

Unable to find image 'hello-world:latest' locally

latest: Pulling from library/hello-world

1b930d010525: Pull complete

Digest: sha256:f9dfddf63636d84ef479d645ab5885156ae030f611a56f3a7ac7f2fdd86d7e4e

Status: Downloaded newer image for hello-world:latest

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.

2. The Docker daemon pulled the "hello-world" image from the Docker Hub.

(amd64)

3. The Docker daemon created a new container from that image which runs the

executable that produces the output you are currently reading.

4. The Docker daemon streamed that output to the Docker client, which sent it

to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:

https://hub.docker.com/

For more examples and ideas, visit:

https://docs.docker.com/get-started/

[node1] (local) root@192.168.0.23 ~

$ docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

0d9481bc98b4 hello-world "/hello" 43 seconds ago Exited (0) 41 seconds ago ecstatic\_bohr

144ed978a896 mongo:latest "docker-entrypoint.s…" 2 minutes ago Up 2 minutes 27017/tcp pwd\_mongo.1.vh0zgtajgad7fca3pspk0cvyh

6c30a3374818 mongo-express:latest "tini -- /docker-ent…" 2 minutes ago Up 2 minutes 8081/tcp pwd\_mongo-express.1.pecqu5ykf9ztc9cwnj7c4lcaa

[node1] (local) root@192.168.0.23 ~

$ docker container run -idt alpine uptime

Unable to find image 'alpine:latest' locally

latest: Pulling from library/alpine

aad63a933944: Already exists

Digest: sha256:b276d875eeed9c7d3f1cfa7edb06b22ed22b14219a7d67c52c56612330348239

Status: Downloaded newer image for alpine:latest

69d33df5109a13ff5008103ce58b55e9f5bbb6900fc2d4f4b60c2dbea4dd3490

[node1] (local) root@192.168.0.23 ~

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

144ed978a896 mongo:latest "docker-entrypoint.s…" 3 minutes ago Up 3 minutes 27017/tcp pwd\_mongo.1.vh0zgtajgad7fca3pspk0cvyh

6c30a3374818 mongo-express:latest "tini -- /docker-ent…" 3 minutes ago Up 3 minutes 8081/tcp pwd\_mongo-express.1.pecqu5ykf9ztc9cwnj7c4lcaa

[node1] (local) root@192.168.0.23 ~

$ docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

69d33df5109a alpine "uptime" 40 seconds ago Exited (0) 39 seconds ago friendly\_brahmagupta

0d9481bc98b4 hello-world "/hello" About a minute ago Exited (0) About a minute ago ecstatic\_bohr

144ed978a896 mongo:latest "docker-entrypoint.s…" 3 minutes ago Up 3 minutes 27017/tcp pwd\_mongo.1.vh0zgtajgad7fca3pspk0cvyh

6c30a3374818 mongo-express:latest "tini -- /docker-ent…" 3 minutes ago Up 3 minutes 8081/tcp pwd\_mongo-express.1.pecqu5ykf9ztc9cwnj7c4lcaa

[node1] (local) root@192.168.0.23 ~

$ docker rename pwd\_mongo-express.1

"docker rename" requires exactly 2 arguments.

See 'docker rename --help'.

Usage: docker rename CONTAINER NEW\_NAME

Rename a container

[node1] (local) root@192.168.0.23 ~

$ docker logs 6c30a3374818

Waiting for mongo:27017...

/docker-entrypoint.sh: line 14: mongo: Name does not resolve

/docker-entrypoint.sh: line 14: /dev/tcp/mongo/27017: Invalid argument

Mon Mar 30 00:19:24 UTC 2020 retrying to connect to mongo:27017 (2/5)

/docker-entrypoint.sh: connect: Connection refused

/docker-entrypoint.sh: line 14: /dev/tcp/mongo/27017: Connection refused

Mon Mar 30 00:19:25 UTC 2020 retrying to connect to mongo:27017 (3/5)

/docker-entrypoint.sh: connect: Connection refused

/docker-entrypoint.sh: line 14: /dev/tcp/mongo/27017: Connection refused

Mon Mar 30 00:19:26 UTC 2020 retrying to connect to mongo:27017 (4/5)

/docker-entrypoint.sh: connect: Connection refused

/docker-entrypoint.sh: line 14: /dev/tcp/mongo/27017: Connection refused

Mon Mar 30 00:19:27 UTC 2020 retrying to connect to mongo:27017 (5/5)

Welcome to mongo-express

------------------------

Mongo Express server listening at http://0.0.0.0:8081

Server is open to allow connections from anyone (0.0.0.0)

basicAuth credentials are "admin:pass", it is recommended you change this in your config.js!

Database connected

Admin Database connected

[node1] (local) root@192.168.0.23 ~

$ docker logs 144ed978a896

about to fork child process, waiting until server is ready for connections.

forked process: 28

2020-03-30T00:19:23.379+0000 I CONTROL [main] \*\*\*\*\* SERVER RESTARTED \*\*\*\*\*

2020-03-30T00:19:23.385+0000 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'

2020-03-30T00:19:23.389+0000 W ASIO [main] No TransportLayer configured during NetworkInterface startup

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] MongoDB starting : pid=28 port=27017 dbpath=/data/db 64-bit host=144ed978a896

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] db version v4.2.5

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] git version: 2261279b51ea13df08ae708ff278f0679c59dc32

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] OpenSSL version: OpenSSL 1.1.1 11 Sep 2018

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] allocator: tcmalloc

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] modules: none

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] build environment:

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] distmod: ubuntu1804

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] distarch: x86\_64

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] target\_arch: x86\_64

2020-03-30T00:19:23.390+0000 I CONTROL [initandlisten] options: { net: { bindIp: "127.0.0.1", port: 27017, tls: { mode: "disabled" } }, processManagement: { fork: true, pidFilePath: "/tmp/docker-entrypoint-temp-mongod.pid" }, systemLog: { destination: "file", logAppend: true, path: "/proc/1/fd/1" } }

2020-03-30T00:19:23.394+0000 I STORAGE [initandlisten] wiredtiger\_open config: create,cache\_size=15567M,cache\_overflow=(file\_max=0M),session\_max=33000,eviction=(threads\_min=4,threads\_max=4),config\_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file\_manager=(close\_idle\_time=100000,close\_scan\_interval=10,close\_handle\_minimum=250),statistics\_log=(wait=0),verbose=[recovery\_progress,checkpoint\_progress],

2020-03-30T00:19:24.152+0000 I STORAGE [initandlisten] WiredTiger message [1585527564:152308][28:0x7f8c66ff6b00], txn-recover: Set global recovery timestamp: (0, 0)

2020-03-30T00:19:24.157+0000 I RECOVERY [initandlisten] WiredTiger recoveryTimestamp. Ts: Timestamp(0, 0)

2020-03-30T00:19:24.162+0000 I STORAGE [initandlisten] Timestamp monitor starting

2020-03-30T00:19:24.163+0000 I CONTROL [initandlisten]

2020-03-30T00:19:24.163+0000 I CONTROL [initandlisten] \*\* WARNING: Access control is not enabled for the database.

2020-03-30T00:19:24.163+0000 I CONTROL [initandlisten] \*\* Read and write access to data and configuration is unrestricted.

2020-03-30T00:19:24.163+0000 I CONTROL [initandlisten]

2020-03-30T00:19:24.165+0000 I STORAGE [initandlisten] createCollection: admin.system.version with provided UUID: acb8432e-3eea-43d1-854e-feb36c8209e9 and options: { uuid: UUID("acb8432e-3eea-43d1-854e-feb36c8209e9") }

2020-03-30T00:19:24.169+0000 I INDEX [initandlisten] index build: done building index \_id\_ on ns admin.system.version

2020-03-30T00:19:24.169+0000 I SHARDING [initandlisten] Marking collection admin.system.version as collection version: <unsharded>

2020-03-30T00:19:24.169+0000 I COMMAND [initandlisten] setting featureCompatibilityVersion to 4.2

2020-03-30T00:19:24.170+0000 I SHARDING [initandlisten] Marking collection local.system.replset as collection version: <unsharded>

2020-03-30T00:19:24.170+0000 I STORAGE [initandlisten] Flow Control is enabled on this deployment.

2020-03-30T00:19:24.170+0000 I SHARDING [initandlisten] Marking collection admin.system.roles as collection version: <unsharded>

2020-03-30T00:19:24.170+0000 I STORAGE [initandlisten] createCollection: local.startup\_log with generated UUID: bc0c2115-0399-417c-ab49-18aaee102018 and options: { capped: true, size: 10485760 }

2020-03-30T00:19:24.174+0000 I INDEX [initandlisten] index build: done building index \_id\_ on ns local.startup\_log

2020-03-30T00:19:24.174+0000 I SHARDING [initandlisten] Marking collection local.startup\_log as collection version: <unsharded>

2020-03-30T00:19:24.174+0000 I FTDC [initandlisten] Initializing full-time diagnostic data capture with directory '/data/db/diagnostic.data'

2020-03-30T00:19:24.176+0000 I SHARDING [LogicalSessionCacheRefresh] Marking collection config.system.sessions as collection version: <unsharded>

2020-03-30T00:19:24.177+0000 I NETWORK [listener] Listening on /tmp/mongodb-27017.sock

2020-03-30T00:19:24.177+0000 I NETWORK [listener] Listening on 127.0.0.1

2020-03-30T00:19:24.177+0000 I NETWORK [listener] waiting for connections on port 27017

2020-03-30T00:19:24.177+0000 I CONTROL [LogicalSessionCacheReap] Sessions collection is not set up; waiting until next sessions reap interval: config.system.sessions does not exist

2020-03-30T00:19:24.177+0000 I STORAGE [LogicalSessionCacheRefresh] createCollection: config.system.sessions with provided UUID: 29e1181d-8cb6-4fee-acf7-c9847883f866 and options: { uuid: UUID("29e1181d-8cb6-4fee-acf7-c9847883f866") }

child process started successfully, parent exiting

2020-03-30T00:19:24.183+0000 I INDEX [LogicalSessionCacheRefresh] index build: done building index \_id\_ on ns config.system.sessions

2020-03-30T00:19:24.188+0000 I INDEX [LogicalSessionCacheRefresh] index build: starting on config.system.sessions properties: { v: 2, key: { lastUse: 1 }, name: "lsidTTLIndex", ns: "config.system.sessions", expireAfterSeconds: 1800 } using method: Hybrid

2020-03-30T00:19:24.188+0000 I INDEX [LogicalSessionCacheRefresh] build may temporarily use up to 200 megabytes of RAM

2020-03-30T00:19:24.189+0000 I INDEX [LogicalSessionCacheRefresh] index build: collection scan done. scanned 0 total records in 0 seconds

2020-03-30T00:19:24.189+0000 I INDEX [LogicalSessionCacheRefresh] index build: inserted 0 keys from external sorter into index in 0 seconds

2020-03-30T00:19:24.190+0000 I INDEX [LogicalSessionCacheRefresh] index build: done building index lsidTTLIndex on ns config.system.sessions

2020-03-30T00:19:24.253+0000 I NETWORK [listener] connection accepted from 127.0.0.1:43682 #1 (1 connection now open)

2020-03-30T00:19:24.253+0000 I NETWORK [conn1] received client metadata from 127.0.0.1:43682 conn1: { application: { name: "MongoDB Shell" }, driver: { name: "MongoDB Internal Client", version: "4.2.5" }, os: { type: "Linux", name: "Ubuntu", architecture: "x86\_64", version: "18.04" } }

2020-03-30T00:19:24.260+0000 I NETWORK [conn1] end connection 127.0.0.1:43682 (0 connections now open)

2020-03-30T00:19:24.334+0000 I NETWORK [listener] connection accepted from 127.0.0.1:43686 #2 (1 connection now open)

2020-03-30T00:19:24.334+0000 I NETWORK [conn2] received client metadata from 127.0.0.1:43686 conn2: { application: { name: "MongoDB Shell" }, driver: { name: "MongoDB Internal Client", version: "4.2.5" }, os: { type: "Linux", name: "Ubuntu", architecture: "x86\_64", version: "18.04" } }

2020-03-30T00:19:24.380+0000 I SHARDING [conn2] Marking collection admin.system.users as collection version: <unsharded>

2020-03-30T00:19:24.380+0000 I STORAGE [conn2] createCollection: admin.system.users with generated UUID: 2594a4b8-de8d-4aa1-8a41-75fa95d8b3a4 and options: {}

2020-03-30T00:19:24.384+0000 I INDEX [conn2] index build: done building index \_id\_ on ns admin.system.users

2020-03-30T00:19:24.387+0000 I INDEX [conn2] index build: done building index user\_1\_db\_1 on ns admin.system.users

Successfully added user: {

"user" : "root",

"roles" : [

{

"role" : "root",

"db" : "admin"

}

]

}

2020-03-30T00:19:24.388+0000 E - [main] Error saving history file: FileOpenFailed: Unable to open() file /home/mongodb/.dbshell: No such file or directory

2020-03-30T00:19:24.389+0000 I NETWORK [conn2] end connection 127.0.0.1:43686 (0 connections now open)

/usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/\*

2020-03-30T00:19:24.414+0000 I CONTROL [main] \*\*\*\*\* SERVER RESTARTED \*\*\*\*\*

2020-03-30T00:19:24.416+0000 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'

2020-03-30T00:19:24.421+0000 W ASIO [main] No TransportLayer configured during NetworkInterface startup

killing process with pid: 28

2020-03-30T00:19:24.421+0000 I CONTROL [signalProcessingThread] got signal 15 (Terminated), will terminate after current cmd ends

2020-03-30T00:19:24.422+0000 I NETWORK [signalProcessingThread] shutdown: going to close listening sockets...

2020-03-30T00:19:24.422+0000 I NETWORK [listener] removing socket file: /tmp/mongodb-27017.sock

2020-03-30T00:19:24.422+0000 I - [signalProcessingThread] Stopping further Flow Control ticket acquisitions.

2020-03-30T00:19:24.422+0000 I CONTROL [signalProcessingThread] Shutting down free monitoring

2020-03-30T00:19:24.422+0000 I FTDC [signalProcessingThread] Shutting down full-time diagnostic data capture

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] Deregistering all the collections

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] Timestamp monitor shutting down

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] WiredTigerKVEngine shutting down

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] Shutting down session sweeper thread

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] Finished shutting down session sweeper thread

2020-03-30T00:19:24.423+0000 I STORAGE [signalProcessingThread] Shutting down journal flusher thread

2020-03-30T00:19:24.458+0000 I STORAGE [signalProcessingThread] Finished shutting down journal flusher thread

2020-03-30T00:19:24.458+0000 I STORAGE [signalProcessingThread] Shutting down checkpoint thread

2020-03-30T00:19:24.458+0000 I STORAGE [signalProcessingThread] Finished shutting down checkpoint thread

2020-03-30T00:19:24.470+0000 I STORAGE [signalProcessingThread] shutdown: removing fs lock...

2020-03-30T00:19:24.470+0000 I CONTROL [signalProcessingThread] now exiting

2020-03-30T00:19:24.470+0000 I CONTROL [signalProcessingThread] shutting down with code:0

MongoDB init process complete; ready for start up.

2020-03-30T00:19:25.452+0000 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'

2020-03-30T00:19:25.455+0000 W ASIO [main] No TransportLayer configured during NetworkInterface startup

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] MongoDB starting : pid=1 port=27017 dbpath=/data/db 64-bit host=144ed978a896

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] db version v4.2.5

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] git version: 2261279b51ea13df08ae708ff278f0679c59dc32

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] OpenSSL version: OpenSSL 1.1.1 11 Sep 2018

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] allocator: tcmalloc

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] modules: none

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] build environment:

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] distmod: ubuntu1804

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] distarch: x86\_64

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] target\_arch: x86\_64

2020-03-30T00:19:25.455+0000 I CONTROL [initandlisten] options: { net: { bindIp: "\*" }, security: { authorization: "enabled" } }

2020-03-30T00:19:25.455+0000 I STORAGE [initandlisten] Detected data files in /data/db created by the 'wiredTiger' storage engine, so setting the active storage engine to 'wiredTiger'.

2020-03-30T00:19:25.456+0000 I STORAGE [initandlisten] wiredtiger\_open config: create,cache\_size=15567M,cache\_overflow=(file\_max=0M),session\_max=33000,eviction=(threads\_min=4,threads\_max=4),config\_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file\_manager=(close\_idle\_time=100000,close\_scan\_interval=10,close\_handle\_minimum=250),statistics\_log=(wait=0),verbose=[recovery\_progress,checkpoint\_progress],

2020-03-30T00:19:26.203+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:203627][1:0x7fcb830fdb00], txn-recover: Recovering log 1 through 2

2020-03-30T00:19:26.298+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:298738][1:0x7fcb830fdb00], txn-recover: Recovering log 2 through 2

2020-03-30T00:19:26.390+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:390521][1:0x7fcb830fdb00], txn-recover: Main recovery loop: starting at 1/29056 to 2/256

2020-03-30T00:19:26.509+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:509352][1:0x7fcb830fdb00], txn-recover: Recovering log 1 through 2

2020-03-30T00:19:26.577+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:577190][1:0x7fcb830fdb00], txn-recover: Recovering log 2 through 2

2020-03-30T00:19:26.635+0000 I STORAGE [initandlisten] WiredTiger message [1585527566:635383][1:0x7fcb830fdb00], txn-recover: Set global recovery timestamp: (0, 0)

2020-03-30T00:19:26.640+0000 I RECOVERY [initandlisten] WiredTiger recoveryTimestamp. Ts: Timestamp(0, 0)

2020-03-30T00:19:26.644+0000 I STORAGE [initandlisten] Timestamp monitor starting

2020-03-30T00:19:26.647+0000 I SHARDING [initandlisten] Marking collection local.system.replset as collection version: <unsharded>

2020-03-30T00:19:26.649+0000 I STORAGE [initandlisten] Flow Control is enabled on this deployment.

2020-03-30T00:19:26.649+0000 I SHARDING [initandlisten] Marking collection admin.system.roles as collection version: <unsharded>

2020-03-30T00:19:26.649+0000 I SHARDING [initandlisten] Marking collection admin.system.version as collection version: <unsharded>

2020-03-30T00:19:26.650+0000 I SHARDING [initandlisten] Marking collection local.startup\_log as collection version: <unsharded>

2020-03-30T00:19:26.650+0000 I FTDC [initandlisten] Initializing full-time diagnostic data capture with directory '/data/db/diagnostic.data'

2020-03-30T00:19:26.651+0000 I SHARDING [LogicalSessionCacheRefresh] Marking collection config.system.sessions as collection version: <unsharded>

2020-03-30T00:19:26.652+0000 I NETWORK [listener] Listening on /tmp/mongodb-27017.sock

2020-03-30T00:19:26.652+0000 I NETWORK [listener] Listening on 0.0.0.0

2020-03-30T00:19:26.652+0000 I NETWORK [listener] waiting for connections on port 27017

2020-03-30T00:19:26.652+0000 I SHARDING [LogicalSessionCacheReap] Marking collection config.transactions as collection version: <unsharded>

2020-03-30T00:19:27.000+0000 I SHARDING [ftdc] Marking collection local.oplog.rs as collection version: <unsharded>

2020-03-30T00:19:27.326+0000 I NETWORK [listener] connection accepted from 10.0.0.4:47082 #1 (1 connection now open)

2020-03-30T00:19:27.326+0000 I NETWORK [conn1] end connection 10.0.0.4:47082 (0 connections now open)

2020-03-30T00:19:27.751+0000 I NETWORK [listener] connection accepted from 10.0.0.4:47084 #2 (1 connection now open)

2020-03-30T00:19:27.757+0000 I NETWORK [conn2] received client metadata from 10.0.0.4:47084 conn2: { driver: { name: "nodejs", version: "2.2.24" }, os: { type: "Linux", name: "linux", architecture: "x64", version: "4.4.0-174-generic" }, platform: "Node.js v12.16.1, LE, mongodb-core: 2.1.8" }

2020-03-30T00:19:27.763+0000 I SHARDING [conn2] Marking collection admin.system.users as collection version: <unsharded>

2020-03-30T00:19:28.189+0000 I ACCESS [conn2] Successfully authenticated as principal root on admin from client 10.0.0.4:47084

2020-03-30T00:19:28.201+0000 I NETWORK [listener] connection accepted from 10.0.0.4:47090 #3 (2 connections now open)

2020-03-30T00:19:28.276+0000 I ACCESS [conn3] Successfully authenticated as principal root on admin from client 10.0.0.4:47090

[node1] (local) root@192.168.0.23 ~

$ docker pull nginx

Using default tag: latest

latest: Pulling from library/nginx

68ced04f60ab: Pull complete

28252775b295: Pull complete

a616aa3b0bf2: Pull complete

Digest: sha256:2539d4344dd18e1df02be842ffc435f8e1f699cfc55516e2cf2cb16b7a9aea0b

Status: Downloaded newer image for nginx:latest

docker.io/library/nginx:latest

[node1] (local) root@192.168.0.23 ~

$ docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mongo latest c5e5843d9f5f 3 days ago 387MB

mongo-express latest a36d72e09c39 5 days ago 127MB

alpine latest a187dde48cd2 6 days ago 5.6MB

nginx latest 6678c7c2e56c 3 weeks ago 127MB

hello-world latest fce289e99eb9 15 months ago 1.84kB

[node1] (local) root@192.168.0.23 ~

$ docker run -it nginx /bin/bash

root@a0f6e33d9cf6:/# ls

bin dev home lib64 mnt proc run srv tmp var

boot etc lib media opt root sbin sys usr

root@a0f6e33d9cf6:/# pwd

/

root@a0f6e33d9cf6:/# exit

exit

[node1] (local) root@192.168.0.23 ~

$ docker run -d -p 8080:08 --name mynginx nginx

bb21f7720d12d9ed68d752f1a8a3bc5ba060c909c600d4cadfdbd05de71f6913

[node1] (local) root@192.168.0.23 ~

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

bb21f7720d12 nginx "nginx -g 'daemon of…" 12 seconds ago Up 11 seconds 80/tcp, 0.0.0.0:8080->8/tcp mynginx

144ed978a896 mongo:latest "docker-entrypoint.s…" 11 minutes ago Up 11 minutes 27017/tcp pwd\_mongo.1.vh0zgtajgad7fca3pspk0cvyh

6c30a3374818 mongo-express:latest "tini -- /docker-ent…" 11 minutes ago Up 11 minutes 8081/tcp pwd\_mongo-express.1.pecqu5ykf9ztc9cwnj7c4lcaa

[node1] (local) root@192.168.0.23 ~

$ docker exec -it mynginx /bin/bash

root@bb21f7720d12:/# cat /ect/os-release

cat: /ect/os-release: No such file or directory

root@bb21f7720d12:/# exit

exit

[node1] (local) root@192.168.0.23 ~

$ docker ps -l

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

bb21f7720d12 nginx "nginx -g 'daemon of…" 2 minutes ago Up 2 minutes 80/tcp, 0.0.0.0:8080->8/tcp mynginx

[node1] (local) root@192.168.0.23 ~

$