Cahier TPs Docker

Table des matières

1 - Installation

Lab 1-1 : Installation sur CentOS

1/ Mettre à jour Centos

```
[root@centos1 ~]# yum update -y
Modules complémentaires chargés : fastestmirror
Loading mirror speeds from cached hostfile
 * base: fr.mirror.babylon.network
 * extras: centos.mirrors.ovh.net
 * updates: centos.mirrors.ovh.net
No packages marked for update
```

2/ Installer les utilitaires yum

```
[root@centos1 ~]# yum install -y yum-utils
Modules complémentaires chargés : fastestmirror
Loading mirror speeds from cached hostfile
 * base: fr.mirror.babylon.network
 * extras: fr.mirror.babylon.network
 * updates: centos.mirrors.ovh.net
Le paquet yum-utils-1.1.31-40.el7.noarch est déjà installé dans sa
dernière version
Rien à faire
```

3/ Ajouter le dépôt docker

```
[root@centos1 ~]# yum-config-manager –add-repo \
      https://download.docker.com/linux/centos/docker-ce.repo
Modules complémentaires chargés : fastestmirror
adding repo from: https://download.docker.com/linux/centos/docker-
ce.repo
grabbing file https://download.docker.com/linux/centos/docker-ce.repo to
/etc/yum.repos.d/docker-ce.repo
repo saved to /etc/yum.repos.d/docker-ce.repo
```

4/ Créer le cache des metadonnées

```
[root@centos1 ~]# yum makecache fast
Modules complémentaires chargés : fastestmirror
base                                                                    |
3.6 kB  00:00:00
docker-ce-stable                                                        |
2.9 kB  00:00:00
extras                                                                  |
3.4 kB  00:00:00
updates                                                                 |
3.4 kB  00:00:00
```

```
docker-ce-stable/x86_64/primary_db                                          |
4.8 kB  00:00:00
Loading mirror speeds from cached hostfile
 * base: fr.mirror.babylon.network
 * extras: centos.mirrors.ovh.net
 * updates: centos.mirrors.ovh.net
Cache des méta données créé
```

4/ Installer docker CE

```
[root@centos1 ~]# yum install docker-ce
Modules complémentaires chargés : fastestmirror
Loading mirror speeds from cached hostfile
 * base: fr.mirror.babylon.network
 * extras: fr.mirror.babylon.network
 * updates: centos.mirrors.ovh.net
Résolution des dépendances
--> Lancement de la transaction de test
---> Le paquet docker-ce.x86_64 0:17.03.1.ce-1.el7.centos sera installé

.../...

Installé :
  docker-ce.x86_64 0:17.03.1.ce-1.el7.centos

Dépendances installées :
  audit-libs-python.x86_64 0:2.6.5-3.el7_3.1            checkpolicy.x86_64 0:2.5-4.el7
  docker-ce-selinux.noarch 0:17.03.1.ce-1.el7.centos    libcgroup.x86_64 0:0.41-11.el7
  libseccomp.x86_64 0:2.3.1-2.el7                       libselinux-python.x86_64 0:2.5-6.el7
  libsemanage-python.x86_64 0:2.5-5.1.el7_3             libtool-ltdl.x86_64 0:2.4.2-22.el7_3
  policycoreutils-python.x86_64 0:2.5-11.el7_3          python-IPy.noarch 0:0.75-6.el7
  setools-libs.x86_64 0:3.3.8-1.1.el7

Terminé !
```

5/ Démarrer et vérifier l'installation de docker

```
[root@centos1 ~]# systemctl start docker

[root@centos1 ~]# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
78445dd45222: Pull complete
Digest:
sha256:c5515758d4c5e1e838e9cd307f6c6a0d620b5e07e6f927b07d05f6d12a1ac8d7
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.
 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:
```

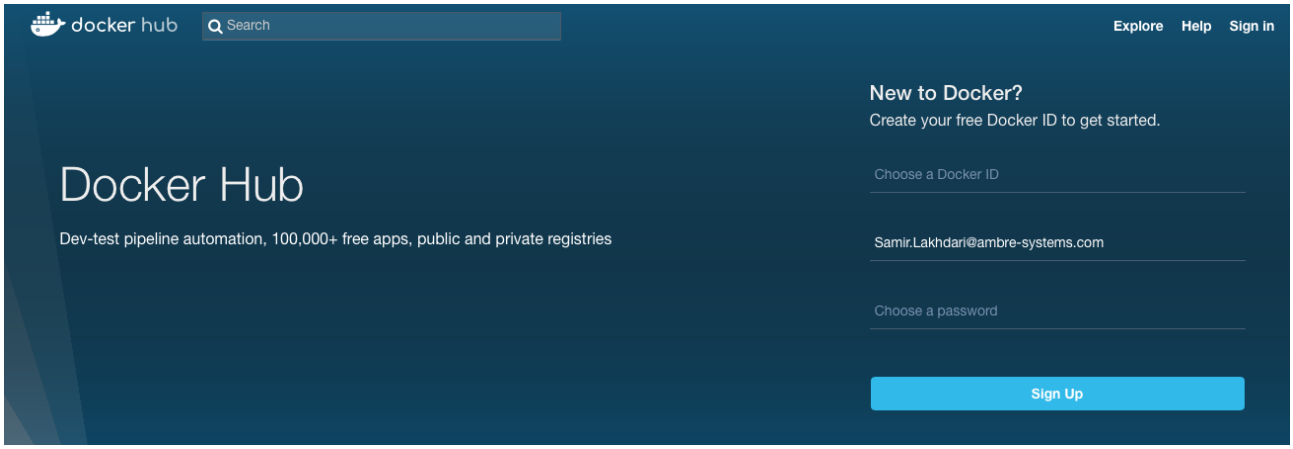
```
https://cloud.docker.com/
```

6/ afficher la version de docker

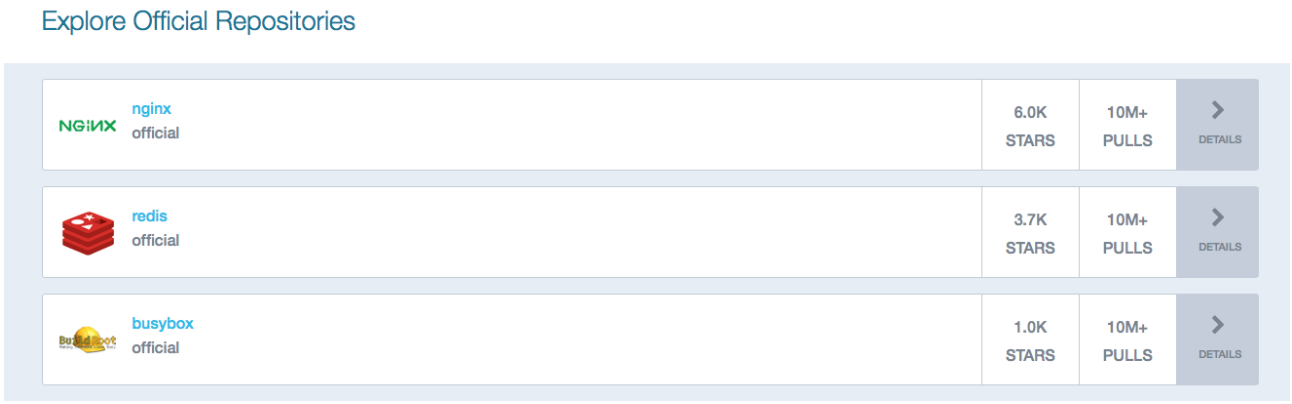
```
[root@centos1 ~]# docker version
Client:
 Version:      17.03.1-ce
 API version:  1.27
 Go version:   go1.7.5
 Git commit:   c6d412e
 Built:        Mon Mar 27 17:05:44 2017
 OS/Arch:      linux/amd64

Server:
 Version:      17.03.1-ce
 API version:  1.27 (minimum version 1.12)
 Go version:   go1.7.5
 Git commit:   c6d412e
 Built:        Mon Mar 27 17:05:44 2017
 OS/Arch:      linux/amd64
 Experimental: false
```

Lab 1-2 : Docker HUB

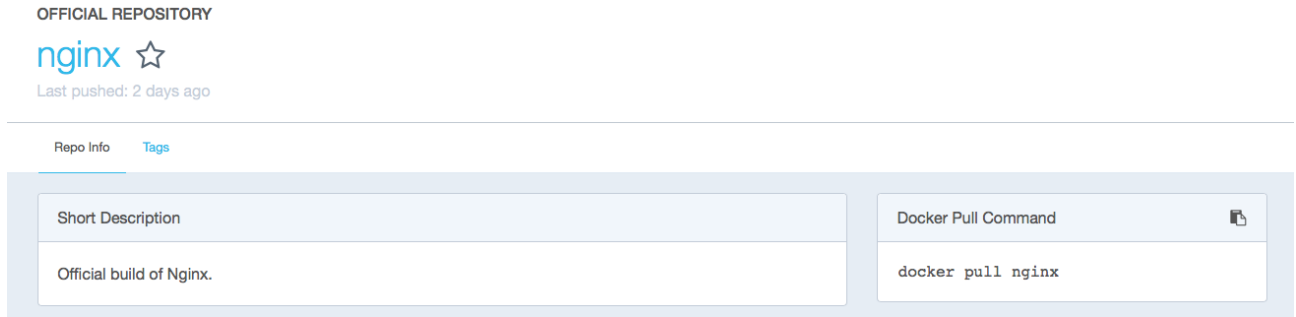
1/ Créer un compte sur le site https://hub.docker.com



2/ Cliquer sur `Explore` pour parcourir les repository officiels



3/ Cliquer sur `DETAILS` pour plus d'informations



Lab 1-3 : Images

1/ Télécharger une image à partir du HUB Docke

```
[root@centos1 ~]# docker pull ubuntu:14.04
14.04: Pulling from library/ubuntu
8f229c550c2e: Pull complete
8e1fb71e8df6: Pull complete
f75a34586856: Pull complete
8744e322b832: Pull complete
d5165bfce78f: Pull complete
Digest:
sha256:edf05697d8ea17028a69726b4b450ad48da8b29884cd640fec950c904bfb50ce
Status: Downloaded newer image for ubuntu:14.04


[root@centos1 ~]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
SIZE
ubuntu              14.04               302fa07d8117        4 weeks ago
188 MB
hello-world         latest              48b5124b2768        4 months ago
1.84 kB
```

2 - Containers

Lab 2-1 : Exécuter un container

1/ Exécuter un container à partir de l'image ubuntu

```
[root@centos1 ~]# docker run ubuntu:14.04 echo "hello world"
hello world
```

2/ Exécuter cette commande et observer le PID de la commande ps

```
[root@centos1 ~]# docker run ubuntu:14.04 ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 09:25 ?        00:00:00 ps -ef
```

3/ Lister les containers en exécution

```
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS               NAMES
```

3/ Lister tous les containers

```
[root@centos1 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS                      PORTS               NAMES
329a4a4265d4        ubuntu:14.04        "ps -ef"                 2 minutes
ago       Exited (0) 2 minutes ago
ecstatic_mirzakhani
4cce702d15ad        ubuntu:14.04        "echo 'hello world'"     4 minutes
ago       Exited (0) 4 minutes ago                       naughty_bardeen
12ff38e588fe        hello-world         "/hello"                 7 minutes
ago       Exited (0) 7 minutes ago
upbeat_brahmagupta
```

Lab 2-2 : Accès au terminal d'un container

1/ Créez un conteneur à l'aide de l'image ubuntu 14.04 et connectez-vous au terminal

```
[root@centos1 ~]# docker run -it ubuntu:14.04 bash
root@2789725e7f65:/#
```

2/ Créez un fichier dans le conteneur puis sortez du conteneur

```
root@2789725e7f65:/# touch fic1

root@2789725e7f65:/# ls
bin  boot  dev  etc  fic1  home  lib  lib64  media  mnt  opt  proc  root
run  sbin  srv  sys  tmp  usr  var

root@2789725e7f65:/# exit
exit
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS               NAMES
```

3/ Exécutez une deuxième fis la commande :

```
[root@centos1 ~]# docker run -it ubuntu:14.04 bash

root@eafed0ffc27e:/# ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt     proc  root
run  sbin  srv  sys  tmp  usr  var
root@eafed0ffc27e:/#
```

Que s'est-il passé ?

Lab 2-3 : Exécution en mode détaché

1/ Exécutez la commande suivante :

```
[root@centos1 ~]# docker run -d centos ping 127.0.0.1 -c 60
Unable to find image 'centos:latest' locally
latest: Pulling from library/centos
Digest:
sha256:bba1de7c9d900a898e3cadbae040dfe8a633c06bc104a0df76ae24483e03c077
Status: Downloaded newer image for centos:latest
2daefc5740000d7d026733baf3d74614caf8415eb1aa604451cfea6da7a6bf4f

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS               NAMES
2daefc574000        centos              "ping 127.0.0.1 -c 60"   5
seconds ago       Up 4 seconds
wizardly_colden
```

2/ Attendez quelques secondes puis lister les contenuers :

```
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS               NAMES
```

3/ Exécutez la commande suivante pour lancer un serveur web :

```
[root@centos1 ~]# docker run -d -P nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
ff3d52d8f55f: Pull complete
b05436c68d6a: Pull complete
961dd3f5d836: Pull complete
Digest:
sha256:12d30ce421ad530494d588f87b2328ddc3cae666e77ea1ae5ac3a6661e52cde6
Status: Downloaded newer image for nginx:latest
02ee4d8a480a874f477f0a90da19a521b43bd7ab9ce9b129b9befdc4bc5bf81a

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                   NAMES
02ee4d8a480a        nginx               "nginx -g 'daemon ..."   5
seconds ago       Up 5 seconds        0.0.0.0:32768->80/tcp
gallant_knuth
```

3/ Notez le mappage de port et allez tester le serveur web avec votre navigateur :

Lab 2-4 : S'attacher à un conteneur et se détacher d'un conteneur

1/ Exécutez la commande suivante :

```
[root@centos1 ~]# docker run -d centos ping 127.0.0.1 -c 60
4b4b1dab96d2021429a80ece3a18cb222feaaf7f8895157884f601adc6c3b91b

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                   NAMES
4b4b1dab96d2        centos              "ping 127.0.0.1 -c 60"   13
seconds ago      Up 13 seconds
quirky_sammet

[root@centos1 ~]# docker attach 4b4b1dab96d2
64 bytes from 127.0.0.1: icmp_seq=26 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=27 ttl=64 time=0.057 ms
64 bytes from 127.0.0.1: icmp_seq=28 ttl=64 time=0.057 ms
64 bytes from 127.0.0.1: icmp_seq=29 ttl=64 time=0.053 ms
64 bytes from 127.0.0.1: icmp_seq=30 ttl=64 time=0.054 ms
```

Appuyez sur CTRL-P-Q pour se détacher du conteneur, que se passe-t-il ?

2/ Exécutez la commande suivante :

```
[root@centos1 ~]# docker run -d -it centos ping 127.0.0.1 -c 60
1e1114dae615b92e1e79ba91fcfacb034b609036d7b65452b693a906b1ad3305

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                   NAMES
1e1114dae615        centos              "ping 127.0.0.1 -c 60"   4
seconds ago       Up 4 seconds
goofy_snyder

[root@centos1 ~]# docker attach 1e1114dae615
64 bytes from 127.0.0.1: icmp_seq=18 ttl=64 time=0.056 ms
64 bytes from 127.0.0.1: icmp_seq=19 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=20 ttl=64 time=0.055 ms
64 bytes from 127.0.0.1: icmp_seq=21 ttl=64 time=0.051 ms
64 bytes from 127.0.0.1: icmp_seq=22 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=23 ttl=64 time=0.050 ms
64 bytes from 127.0.0.1: icmp_seq=24 ttl=64 time=0.049 ms
```

Appuyez sur CTRL-P-Q pour se détacher du conteneur

```
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
```

```
STATUS              PORTS                    NAMES
1e1114dae615        centos              "ping 127.0.0.1 -c 60"   28
seconds ago      Up 27 seconds
goofy_snyder
```

Lab 2-5 : La commande exec

1/ Exécutez un conteneur en mode arrière plan :

```
[root@centos1 ~]# docker run -d -it centos bash
9d5a92f0a6e47006d9b8cfec89dcb12cb1dea5f657055674202993b916a42c1e

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS               NAMES
9d5a92f0a6e4        centos              "bash"                   5
seconds ago       Up 4 seconds
stupefied_babbage

[root@centos1 ~]# docker attach 9d5a92f0a6e4
[root@9d5a92f0a6e4 /]# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 10:13 ?        00:00:00 bash
root        13     1  0 10:13 ?        00:00:00 ps -ef
[root@9d5a92f0a6e4 /]#
```

Appuyez sur CTRL-P-Q pour se détacher du conteneur

```
[root@centos1 ~]#
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS               NAMES
9d5a92f0a6e4        centos              "bash"                   29
seconds ago      Up 29 seconds
stupefied_babbage

[root@centos1 ~]# docker exec -it 9d5a92f0a6e4 bash
[root@9d5a92f0a6e4 /]# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 10:13 ?        00:00:00 bash
root        18     0  0 10:14 ?        00:00:00 bash
root        30    18  0 10:14 ?        00:00:00 ps -ef
[root@9d5a92f0a6e4 /]#
[root@9d5a92f0a6e4 /]# exit
```

Remarquez les PPID

Lab 2-6 : La commande `logs`

1/ Exécutez un conteneur en mode arrière plan :

```
[root@centos1 ~]# docker run -d ubuntu:14.04 ping 127.0.0.1 -c 100
4ca6589443712b47acececa403d9656f132dfbd54302e46cf261841f194cc307
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS               NAMES
4ca658944371        ubuntu:14.04        "ping 127.0.0.1 -c..."   7
seconds ago       Up 7 seconds
friendly_feynman
[root@centos1 ~]# docker logs 4ca658944371
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=6 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=7 ttl=64 time=0.043 ms
64 bytes from 127.0.0.1: icmp_seq=8 ttl=64 time=0.045 ms

[root@centos1 ~]# docker logs -f 4ca658944371
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=6 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=7 ttl=64 time=0.043 ms
64 bytes from 127.0.0.1: icmp_seq=8 ttl=64 time=0.045 ms
64 bytes from 127.0.0.1: icmp_seq=9 ttl=64 time=0.053 ms
^C
[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS               NAMES
4ca658944371        ubuntu:14.04        "ping 127.0.0.1 -c..."   51
seconds ago      Up 50 seconds
friendly_feynman
[root@centos1 ~]# docker logs --tail 10 -f 4ca658944371
64 bytes from 127.0.0.1: icmp_seq=58 ttl=64 time=0.056 ms
64 bytes from 127.0.0.1: icmp_seq=59 ttl=64 time=0.056 ms
64 bytes from 127.0.0.1: icmp_seq=60 ttl=64 time=0.058 ms
64 bytes from 127.0.0.1: icmp_seq=61 ttl=64 time=0.059 ms
```

Lab 2-7 : Les commandes `stop - start`

1/ Exécutez un conteneur en mode arrière plan :

```
[root@centos1 ~]# docker run -d -P nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
ff3d52d8f55f: Pull complete
b05436c68d6a: Pull complete
961dd3f5d836: Pull complete
Digest:
sha256:12d30ce421ad530494d588f87b2328ddc3cae666e77ea1ae5ac3a6661e52cde6
Status: Downloaded newer image for nginx:latest
dfbeb772aac7661129d9bea2ac800363e191b19cd8e3423b43e97fe214a00e14

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                   NAMES
dfbeb772aac7        nginx               "nginx -g 'daemon ..."   3
seconds ago       Up 3 seconds        0.0.0.0:32769->80/tcp
quizzical_swirles

[root@centos1 ~]# curl http://127.0.0.1:32769/
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed
and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

2/ Arrêtez le conteneur :

```
[root@centos1 ~]# docker stop dfbeb772aac7
dfbeb772aac7

[root@centos1 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS                      PORTS               NAMES
dfbeb772aac7        nginx               "nginx -g 'daemon ..."   3
```

```
minutes ago       Exited (0) 6 seconds ago
quizzical_swirles
```

3/ redémarrez le conteneur :

```
[root@centos1 ~]# docker start dfbeb772aac7
dfbeb772aac7

[root@centos1 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                   NAMES
dfbeb772aac7        nginx               "nginx -g 'daemon ..."   6
minutes ago       Up 2 seconds        0.0.0.0:32770->80/tcp
quizzical_swirles
```

Lab 2-8 : La commande `inspect`

1/ La commande `inspect` permet d'afficher les propriétés :

```
[root@centos1 ~]# docker inspect dfbeb772aac7
[
    {
        "Id":
"dfbeb772aac7661129d9bea2ac800363e191b19cd8e3423b43e97fe214a00e14",
        "Created": "2017-05-15T16:57:53.538978324Z",
        "Path": "nginx",
        "Args": [
            "-g",
            "daemon off;"
        ],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 5472,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2017-05-15T17:04:44.374755739Z",
            "FinishedAt": "2017-05-15T17:01:17.479906146Z"
        },
        "Config": {
            "Hostname": "dfbeb772aac7",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [

"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "NGINX_VERSION=1.13.0-1~stretch",
                "NJS_VERSION=1.13.0.0.1.10-1~stretch"
            ],
            "Cmd": [
                "nginx",
                "-g",
                "daemon off;"
```

```
            ],
        "NetworkSettings": {
            "Bridge": "",
            "Ports": {
                "80/tcp": [
                    {
                        "HostIp": "0.0.0.0",
                        "HostPort": "32770"
                    }
                ]
            },
            "Gateway": "172.17.0.1",
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
…/...
```

2/ La commande `inspect` et l'option `-format` :

```
[root@centos1 ~]# docker inspect
--format='{{.NetworkSettings.IPAddress}}' dfbeb772aac7
172.17.0.2

[root@centos1 ~]# docker inspect --format='{{.NetworkSettings}}'
dfbeb772aac7
{{ eef5ce9a601560e24a0b33633ad6848c41b55cb358fce824d1441c888c258bc1
false  0 map[80/tcp:[{0.0.0.0 32770}]]
/var/run/docker/netns/eef5ce9a6015 [] []}
{ee91cd1c31ecdc3be99b1df469a9b6a9668ced9e8ff7b0df4dd78618ec783637
172.17.0.1  0 172.17.0.2 16  02:42:ac:11:00:02}
map[bridge:0xc4200c4900]}

[root@centos1 ~]# docker inspect --format='{{json .NetworkSettings}}'
dfbeb772aac7
{"Bridge":"","SandboxID":"eef5ce9a601560e24a0b33633ad6848c41b55cb358fce8
24d1441c888c258bc1","HairpinMode":false,"LinkLocalIPv6Address":"","LinkL
ocalIPv6PrefixLen":0,"Ports":{"80/tcp":
[{"HostIp":"0.0.0.0","HostPort":"32770"}]},"SandboxKey":"/var/run/docker
/netns/eef5ce9a6015","SecondaryIPAddresses":null,"SecondaryIPv6Addresses
":null,"EndpointID":"ee91cd1c31ecdc3be99b1df469a9b6a9668ced9e8ff7b0df4dd
78618ec783637","Gateway":"172.17.0.1","GlobalIPv6Address":"","GlobalIPv6
PrefixLen":0,"IPAddress":"172.17.0.2","IPPrefixLen":16,"IPv6Gateway":"",
"MacAddress":"02:42:ac:11:00:02","Networks":{"bridge":
{"IPAMConfig":null,"Links":null,"Aliases":null,"NetworkID":"ca1e694382c6
c950adae359bc00befa30b91d891449de5c425f4169549cecb13","EndpointID":"ee91
cd1c31ecdc3be99b1df469a9b6a9668ced9e8ff7b0df4dd78618ec783637","Gateway":
"172.17.0.1","IPAddress":"172.17.0.2","IPPrefixLen":16,"IPv6Gateway":"",
"GlobalIPv6Address":"","GlobalIPv6PrefixLen":0,"MacAddress":"02:42:ac:11
:00:02"}}}
```

Lab 2-9 : Supprimer un conteneur

1/ Listez les conteneurs arrêtés :

```
[root@centos1 ~]# docker ps --filter='status=exited'
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS                      PORTS               NAMES
dfbeb772aac7        nginx               "nginx -g 'daemon ..."   33
minutes ago      Exited (0) 42 seconds ago
quizzical_swirles
```

2/ Supprimez le conteneur arrêté :

```
[root@centos1 ~]# docker rm dfbeb772aac7
dfbeb772aac7
```

3/ Supprimez tous les conteneurs arrêtés :

```
[root@centos1 ~]# docker rm $(docker ps -aq)
6217fa8ecf9b
64a388190e2a
3931d3631805
```

3 - Gestion des images

Lab 3-1 : Modifications dans un container

1/ Exécutez un container à partir de l'image `centos` et installez `wget` :

```
[root@centos1 ~]# docker run -it centos:7 bash
[root@95bb28144c48 /]# yum install wget -y
Loaded plugins: fastestmirror, ovl
Loading mirror speeds from cached hostfile
 * base: fr.mirror.babylon.network
 * extras: fr.mirror.babylon.network
 * updates: centos.mirror.fr.planethoster.net
Resolving Dependencies
--> Running transaction check
---> Package wget.x86_64 0:1.14-13.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
================================================================================
 Package                       Arch                                  Version
Repository                     Size
================================================================================
================================================================================
Installing:
 wget                          x86_64                                1.14-
13.el7                            base                               546 k

Transaction Summary
================================================================================
================================================================================
Install  1 Package

Total download size: 546 k
Installed size: 2.0 M
Downloading packages:
warning: /var/cache/yum/x86_64/7/base/packages/wget-1.14-
13.el7.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID f4a80eb5:
NOKEY
Public key for wget-1.14-13.el7.x86_64.rpm is not installed
wget-1.14-13.el7.x86_64.rpm
| 546 kB  00:00:00
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
Importing GPG key 0xF4A80EB5:
 Userid     : "CentOS-7 Key (CentOS 7 Official Signing Key)
<security@centos.org>"
 Fingerprint: 6341 ab27 53d7 8a78 a7c2 7bb1 24c6 a8a7 f4a8 0eb5
 Package    : centos-release-7-3.1611.el7.centos.x86_64 (@CentOS)
 From       : /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
Running transaction check
Running transaction test
Transaction test succeeded
```

```
Running transaction
  Installing : wget-1.14-13.el7.x86_64
1/1
install-info: No such file or directory for /usr/share/info/wget.info.gz
  Verifying  : wget-1.14-13.el7.x86_64
1/1

Installed:
  wget.x86_64 0:1.14-13.el7

Complete!
```

2/ Comparez le conteneur modifié et l'image de base :

```
[root@centos1 ~]# docker diff 95bb28144c48
C /etc
A /etc/wgetrc
C /root
A /root/.bash_history
A /usr/bin/wget
C /var
C /var/lib
C /var/lib/rpm
C /var/lib/yum
C /var/lib/yum/history
C /var/lib/yum/history/2017-05-10
A /var/lib/yum/history/2017-05-10/3
A /var/lib/yum/history/2017-05-10/3/config-repos
A /var/lib/yum/history/2017-05-10/3/saved_tx
A /var/lib/yum/history/2017-05-10/3/config-main
A /var/lib/yum/history/2017-05-10/4
A /var/lib/yum/history/2017-05-10/4/config-main
A /var/lib/yum/history/2017-05-10/4/config-repos
A /var/lib/yum/history/2017-05-10/4/saved_tx
C /var/lib/yum/history/history-2017-05-10.sqlite
C /var/lib/yum/history/history-2017-05-10.sqlite-journal
C /var/lib/yum/rpmdb-indexes
A /var/lib/yum/rpmdb-indexes/conflicts
A /var/lib/yum/rpmdb-indexes/file-requires
A /var/lib/yum/rpmdb-indexes/obsoletes
A /var/lib/yum/rpmdb-indexes/pkgtups-checksums
A /var/lib/yum/rpmdb-indexes/version
C /var/lib/yum/yumdb
A /var/lib/yum/yumdb/w/062ccdaa45d226931f12106e692458f16fb179e6-wget-
1.14-13.el7-x86_64/ts_install_langs
A /var/lib/yum/yumdb/w/062ccdaa45d226931f12106e692458f16fb179e6-wget-
1.14-13.el7-x86_64/var_infra
A /var/lib/yum/yumdb/w/062ccdaa45d226931f12106e692458f16fb179e6-wget-
1.14-13.el7-x86_64/releasever
C /var/log
C /var/log/yum.log
```

Lab 3-2 : Créer une nouvelle image

1/ Créez une nouvelle image à partir du conteneur modifié :

```
[root@centos1 ~]# docker commit 95bb28144c48 masociete/moncentos:1.0
sha256:1df7429bcb148fd50b9177a40c7296cbbc1ca31c288b9469d8a218b8e41b009e

[root@centos1 ~]# docker images
REPOSITORY            TAG                 IMAGE ID            CREATED
SIZE
masociete/moncentos   1.0                 1df7429bcb14        5 seconds
ago       281 MB
centos                7                   8140d0c64310        3 days ago
193 MB
centos                latest              8140d0c64310        3 days ago
193 MB
nginx                 latest              3448f27c273f        5 days ago
109 MB
ubuntu                14.04               302fa07d8117        4 weeks
ago         188 MB
hello-world           latest              48b5124b2768        4 months
ago        1.84 kB
```

2/ Créez un conteneur à partir de la nouvelle image :

```
[root@centos1 ~]# docker run -it masociete/moncentos:1.0 bash
[root@bf8d4d6074de /]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try `wget --help' for more options.
```

Lab 3-3 : Dockerfile

1/ Créez un fichier Dockerfile dans un nouveau répertoire :

```
[root@centos1 ~]# mkdir moncentos
[root@centos1 ~]# cd moncentos
[root@centos1 moncentos]# vi Dockerfile
FROM centos:7
RUN yum update
RUN yum install wget -y

[root@centos1 moncentos]# docker build -t moncentos .
Sending build context to Docker daemon 2.048 kB
Step 1/3 : FROM centos:7
 ---> 8140d0c64310
Step 2/3 : RUN yum update
 ---> Running in 6a6a584f5b8a
Loaded plugins: fastestmirror, ovl
Determining fastest mirrors
 * base: mirror.neify.es
 * extras: mirrors.ircam.fr
 * updates: mirrors.ircam.fr
No packages marked for update
 ---> 6031f9d5dbbb
Removing intermediate container 6a6a584f5b8a
Step 3/3 : RUN yum install wget -y
 ---> Running in 89c2e2557c76
Loaded plugins: fastestmirror, ovl
Loading mirror speeds from cached hostfile
 * base: mirror.neify.es
 * extras: mirrors.ircam.fr
 * updates: mirrors.ircam.fr
Resolving Dependencies
--> Running transaction check
---> Package wget.x86_64 0:1.14-13.el7 will be installed
--> Finished Dependency Resolution

.../...

Running transaction
  Installing : wget-1.14-13.el7.x86_64
1/1
install-info: No such file or directory for /usr/share/info/wget.info.gz
  Verifying  : wget-1.14-13.el7.x86_64
1/1

Installed:
  wget.x86_64 0:1.14-13.el7

Complete!
 ---> a07cb8ea7398
Removing intermediate container 89c2e2557c76
```

```
Successfully built a07cb8ea7398

[root@centos1 moncentos]# docker images
REPOSITORY            TAG                 IMAGE ID            CREATED
SIZE
moncentos             latest              a07cb8ea7398        5 seconds
ago       282 MB
masociete/moncentos   1.0                 1df7429bcb14        8 minutes
ago       281 MB
centos                7                   8140d0c64310        3 days ago
193 MB
centos                latest              8140d0c64310        3 days ago
193 MB
nginx                 latest              3448f27c273f        5 days ago
109 MB
ubuntu                14.04               302fa07d8117        4 weeks
ago         188 MB
hello-world           latest              48b5124b2768        4 months
ago        1.84 kB
```

2/ Modifiez le fichier Dockerfile et ajoutez l'installation d'un nouveau paquet, remarquez l'utilisation du cache :

```
[root@centos1 moncentos]# cat Dockerfile
FROM centos:7
RUN yum update
RUN yum install wget -y
RUN yum install zip -y

[root@centos1 moncentos]# docker build -t moncentos .
Sending build context to Docker daemon 2.048 kB
Step 1/4 : FROM centos:7
 ---> 8140d0c64310
Step 2/4 : RUN yum update
 ---> Using cache
 ---> 6031f9d5dbbb
Step 3/4 : RUN yum install wget -y
 ---> Using cache
 ---> a07cb8ea7398
Step 4/4 : RUN yum install zip -y
 ---> Running in e21877fea220
Loaded plugins: fastestmirror, ovl
Loading mirror speeds from cached hostfile
 * base: mirror.neify.es
 * extras: mirrors.ircam.fr
 * updates: mirrors.ircam.fr
Resolving Dependencies
--> Running transaction check
---> Package zip.x86_64 0:3.0-11.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
========
 Package        Arch              Version                 Repository
Size
================================================================================
========
Installing:
 zip            x86_64            3.0-11.el7              base
260 k

Transaction Summary
================================================================================
========
Install  1 Package

Total download size: 260 k
Installed size: 796 k
Downloading packages:
```

```
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : zip-3.0-11.el7.x86_64
1/1
  Verifying  : zip-3.0-11.el7.x86_64
1/1

Installed:
  zip.x86_64 0:3.0-11.el7

Complete!
 ---> e6adaba320e0
Removing intermediate container e21877fea220
Successfully built e6adaba320e0
```

3/ Visualisez l'historique de la construction des image :

```
[root@centos1 moncentos]# docker history moncentos
IMAGE               CREATED             CREATED BY
SIZE                COMMENT
e6adaba320e0        7 minutes ago       /bin/sh -c yum install zip -y
20.1 MB
a07cb8ea7398        17 minutes ago      /bin/sh -c yum install wget -y
20 MB
6031f9d5dbbb        17 minutes ago      /bin/sh -c yum update
69.3 MB
8140d0c64310        3 days ago          /bin/sh -c #(nop)  CMD
["/bin/bash"]            0 B
<missing>           3 days ago          /bin/sh -c #(nop)  LABEL
name=CentOS Base ...   0 B
<missing>           3 days ago          /bin/sh -c #(nop) ADD
file:f3be3f14a2136b0...   193 MB
```

4/ Modifiez le ficher Dockerfile comme ceci, et que remarquez-vous :

```
[root@centos1 moncentos]# cat Dockerfile
FROM centos:7
RUN yum update
RUN yum install wget zip -y

[root@centos1 moncentos]# docker build -t moncentos .
Sending build context to Docker daemon 2.048 kB
Step 1/3 : FROM centos:7
 ---> 8140d0c64310
Step 2/3 : RUN yum update
 ---> Using cache
 ---> 6031f9d5dbbb
Step 3/3 : RUN yum install wget zip -y
 ---> Running in b3afbfe2f833
Loaded plugins: fastestmirror, ovl
Loading mirror speeds from cached hostfile
 * base: mirror.neify.es
 * extras: mirrors.ircam.fr
 * updates: mirrors.ircam.fr
Resolving Dependencies
--> Running transaction check
---> Package wget.x86_64 0:1.14-13.el7 will be installed
---> Package zip.x86_64 0:3.0-11.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package         Arch              Version                 Repository
Size
```

```
================================================================================
========
Installing:
 wget            x86_64            1.14-13.el7            base
546 k
 zip             x86_64            3.0-11.el7             base
260 k

Transaction Summary
================================================================================
========
Install  2 Packages
  Installing : zip-3.0-11.el7.x86_64
1/2
  Installing : wget-1.14-13.el7.x86_64
2/2
Installed:
  wget.x86_64 0:1.14-13.el7               zip.x86_64 0:3.0-11.el7

Complete!
 ---> b7c5fb141e81
Removing intermediate container b3afbfe2f833
Successfully built b7c5fb141e81
```

4 - Gestion des volumes

Lab 3-1 : Création des volumes

1/ Créez un volume `vol1`:

```
[root@centos1 ~]# docker volume  create --name vol1
vol1
```

2/ Listez les volumes:

```
[root@centos1 ~]# docker volume ls
DRIVER              VOLUME NAME
local               vol1
```

3/ Montez le volume `vol1` dans un conteneur centos:

```
[root@centos1 ~]# docker run -it -v vol1:/datas centos bash
[root@4c6dad9a32de /]# df -h
Filesystem               Size  Used Avail Use% Mounted on
overlay                   14G  2.7G   12G  19% /
tmpfs                    920M     0  920M   0% /dev
tmpfs                    920M     0  920M   0% /sys/fs/cgroup
/dev/mapper/centos-root   14G  2.7G   12G  19% /datas
shm                       64M     0   64M   0% /dev/shm
tmpfs                    920M     0  920M   0% /sys/firmware
```

4/ Créez un ficher dans conteneur centos:

```
[root@4c6dad9a32de /]# cat >> /datas/fic1
11111
22222
```

5/ Faites un `exit` pour sortir du conteneur centos:

```
[root@4c6dad9a32de /]# exit
```

6/ Créez une image à partir du conteneur :

```
[root@centos1 ~]# docker commit 4c6dad9a32de mycentos:2.0
sha256:f5599a7dcaf4a2d8ce49d5d783a8f2c9e372cb793f3429a377897a110df4279b
```

7/ Lancez un conteneur à partir de la nouvelle image, que remarquez-vous :

```
[root@centos1 ~]# docker run -it mycentos:2.0 bash
[root@7030d2429bef /]# df
```

```
Filesystem              1K-blocks    Used Available Use% Mounted on
overlay                  14530560 2739588  11790972  19% /
tmpfs                      942064       0    942064   0% /dev
tmpfs                      942064       0    942064   0% /sys/fs/cgroup
/dev/mapper/centos-root  14530560 2739588  11790972  19% /etc/hosts
shm                         65536       0     65536   0% /dev/shm
tmpfs                      942064       0    942064   0% /sys/firmware
```

Lab 3-2 : Utilisation des volumes

1/ Listez les volumes :

```
[root@centos1 ~]# docker volume ls
DRIVER              VOLUME NAME
local               vol1
```

2/ Affichez les caractéristiques d'un volume :

```
[root@centos1 ~]# docker volume inspect vol1
[
    {
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/vol1/_data",
        "Name": "vol1",
        "Options": {},
        "Scope": "local"
    }
]
```

3/ Vérifiez la persistance des données dans le volume :

```
[root@centos1 ~]# cd /var/lib/docker/volumes/vol1/_data
[root@centos1 _data]# ls
fic1
[root@centos1 _data]# cat fic1
11111
22222
```

4/ Lancez un nouveau conteneur et accéder au même volume et créez un autre fichier :

```
[root@centos1 _data]# docker run -it -v vol1:/datas centos bash
[root@f089099d3945 /]# cat /datas/fic1
11111
22222

[root@f089099d3945 /]# cat >> /datas/fic2
abc
def
```

5/ Passez le conteneur en arrière plan par CTRL+P+Q puis connectez vous par un exec :

```
[root@centos1 _data]# docker exec -it  f089099d3945 bash
```

```
[root@f089099d3945 /]# ls /datas
fic1  fic2
```

Lab 3-3 : Suppression des volumes

1/ Listez les volumes :

```
[root@centos1 ~]# docker volume ls
DRIVER              VOLUME NAME
local               vol1
```

2/ Supprimez le volume `vol1` :

```
[root@centos1 ~]# docker volume rm vol1
Error response from daemon: unable to remove volume: remove vol1: volume is in use -
f089099d39450501ed623261447ac22476c4402a8a65bb094924cfb9216ace62

[root@centos1 ~]# docker stop f089099d3945
f089099d3945

[root@centos1 ~]# docker rm f089099d3945
f089099d3945

[root@centos1 ~]# docker volume rm vol1
vol1
```

Lab 3-4 : Montez un volume host

1/ Montez un volume host :

```
[root@centos1 ~]# mkdir html_datas
[root@centos1 ~]# cd html_datas/
[root@centos1 html_datas]# cat >> index.html
Hello On My Web Server

[root@centos1 html_datas]# docker run -it -v /root/html_datas:/data/www
centos
[root@4981354ced38 /]# cd /data/www/
[root@4981354ced38 www]# cat index.html
Hello On My Web Server
[root@4981354ced38 www]# exit
exit
```

2/ Volumes pour les logs :

```
[root@centos1 ~]# docker volume create --name nginx_logs
nginx_logs

[root@centos1 ~]#  docker run -d -P --name nginx_server -v
/root/html_datas:/usr/share/nginx/html -v nginx_logs:/var/log/nginx
nginx
d00640df5f65335c4f511a2135feb030b48a60af530fa661c1931182d6bffe59

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE                  COMMAND
CREATED             STATUS              PORTS
NAMES
d00640df5f65        nginx                  "nginx -g 'daemon ..."   6
seconds ago       Up 6 seconds        0.0.0.0:32770->80/tcp
nginx_server

[root@centos1 ~]# curl http://0.0.0.0:32770
Hello On My Web Server

[root@centos1 ~]# docker exec -it nginx_server bash
root@d00640df5f65:/# cd /var/log/nginx/
root@d00640df5f65:/var/log/nginx# ls
access.log  error.log
root@d00640df5f65:/var/log/nginx# tail -f access.log
```

3/ Inspectez les logs :

```
[root@centos1 ~]# docker volume inspect nginx_logs
[
    {
        "Driver": "local",
        "Labels": {},
```

```
        "Mountpoint": "/var/lib/docker/volumes/nginx_logs/_data",
        "Name": "nginx_logs",
        "Options": {},
        "Scope": "local"
    }
]
```

```
[root@centos1 ~]# cd /var/lib/docker/volumes/nginx_logs/_data
[root@centos1 _data]# ls
access.log  error.log
```

Lab 3-4 : Les volumes dans un Dockerfile

1/ Montez un volume host :

```
[root@centos1 ~]# mkdir html_datas
[root@centos1 ~]# cd html_datas/
[root@centos1 html_datas]# cat >> index.html
Hello On My Web Server

[root@centos1 html_datas]# docker run -it -v /root/html_datas:/data/www
centos
[root@4981354ced38 /]# cd /data/www/
[root@4981354ced38 www]# cat index.html
Hello On My Web Server
[root@4981354ced38 www]# exit
exit
```

5 - Réseau docker

Lab 5-1 : Utilisation du réseau par défaut

1/ Lancez un conteneur en mode daemon :

```
[root@centos1 ~]# docker run --name mycentos1 -d -it centos:7 bash
b7ca2432f21ff213bb7a92ed826ff65e5d067d6377b4cb4da09b81697ddb1e52

[root@centos1 ~]# docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id":
"0a0eaece6487cf24674b19640c6a5e4dff46c7fcc114172e6a3c7ee6a24981f3",
        "Created": "2017-05-18T12:59:03.343980732-04:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Containers": {

"b7ca2432f21ff213bb7a92ed826ff65e5d067d6377b4cb4da09b81697ddb1e52": {
                "Name": "mycentos1",
                "EndpointID":
"11f810809753675a5a7e07e653c3b7eaf2bf76389c1f075cbe06f3dad3289565",
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "IPv6Address": ""
            }
        },
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "com.docker.network.bridge.enable_ip_masquerade": "true",
            "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
            "com.docker.network.bridge.name": "docker0",
            "com.docker.network.driver.mtu": "1500"
        },
        "Labels": {}
    }
]
```

2/ Lancez un autre conteneur et testez le réseau :

```
[root@centos1 ~]# docker run --name mycentos2  -it centos:7 bash
[root@fe73b8e62f72 /]# ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.084 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.087 ms
^C
--- 172.17.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.084/0.085/0.087/0.009 ms
```

```
[root@fe73b8e62f72 /]# ping mycentos1
ping: mycentos1: Name or service not known
```

Lab 5-2 : Création d'un bridge

1/ Créez un bridge :

```
[root@centos1 ~]# docker network  create --driver bridge mybridge
c173166780b2a706bd0d80d1c36816e0f636b12f489640644a3c80e7b39c7e46

[root@centos1 ~]# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
0a0eaece6487        bridge              bridge              local
36d4cba75cb7        host                host                local
c173166780b2        mybridge            bridge              local

[root@centos1 ~]# docker run --name mycentos3 --net=mybridge -d -it
centos:7 bash
eb9b92481282e772379a66c74f5b7723b35583fe180a386b2d7f821ef50a0417

[root@centos1 ~]# docker run --name mycentos4 --net=mybridge -it
centos:7 bash
[root@d24e13f3f36f /]# ping mycentos3
PING mycentos3 (172.21.0.2) 56(84) bytes of data.
64 bytes from mycentos3.mybridge (172.21.0.2): icmp_seq=1 ttl=64
time=0.060 ms
64 bytes from mycentos3.mybridge (172.21.0.2): icmp_seq=2 ttl=64
time=0.088 ms
64 bytes from mycentos3.mybridge (172.21.0.2): icmp_seq=3 ttl=64
time=0.089 ms
64 bytes from mycentos3.mybridge (172.21.0.2): icmp_seq=4 ttl=64
time=0.087 ms
64 bytes from mycentos3.mybridge (172.21.0.2): icmp_seq=5 ttl=64
time=0.085 ms
^C
--- mycentos3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.060/0.081/0.089/0.015 ms
```

Lab 5-2 : Connexion à plusieurs réseaux

1/ Créez un bridge :

```
[root@centos1 ~]# docker exec -it mycentos3 bash
[root@ eb9b92481282 /]# ping mycentos1
ping: mycentos1: Name or service not known


[root@centos1 ~]# docker network connect mybridge mycentos1

[root@centos1 ~]# docker exec -it mycentos3 bash
[root@eb9b92481282 /]# ping mycentos1
PING mycentos1 (172.21.0.3) 56(84) bytes of data.
64 bytes from mycentos1.mybridge (172.21.0.3): icmp_seq=1 ttl=64
time=0.115 ms
64 bytes from mycentos1.mybridge (172.21.0.3): icmp_seq=2 ttl=64
time=0.084 ms
64 bytes from mycentos1.mybridge (172.21.0.3): icmp_seq=3 ttl=64
time=0.085 ms
64 bytes from mycentos1.mybridge (172.21.0.3): icmp_seq=4 ttl=64
time=0.086 ms
^C
--- mycentos1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 0.084/0.092/0.115/0.016 ms
```

Lab 5-3 : Mapping de port

1/ Lancez un conteneur avec mapping manuel :

```
[root@centos1 ~]# docker run -d -p 80:80 -p 90:8080 nginx
cb8a814718547d67f61a13ad00afc52c279514bc109e173c2078006023b04f58

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                                    NAMES
cb8a81471854        nginx               "nginx -g 'daemon ..."   3
seconds ago       Up 2 seconds        0.0.0.0:80->80/tcp, 0.0.0.0:90-
>8080/tcp   jovial_visvesvaraya

[root@centos1 ~]# docker port cb8a81471854
80/tcp -> 0.0.0.0:80
8080/tcp -> 0.0.0.0:90
```

2/ Lancez un conteneur avec mapping automatique :

```
[root@centos1 ~]# docker run -d -P nginx
aa8379b2e9034102ccdda87d76be405d0c484572aeede94816abfe534da17c2d

[root@centos1 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                    NAMES
aa8379b2e903        nginx               "nginx -g 'daemon ..."   4
seconds ago       Up 3 seconds        0.0.0.0:32771->80/tcp
hardcore_jepsen

[root@centos1 ~]# docker port aa8379b2e903
80/tcp -> 0.0.0.0:32771
```

3/ Mapping de port dans Dockerfile :

```
[root@centos1 monimage]# cat Dockerfile
FROM centos:7
RUN yum install -y wget
EXPOSE 80 8080
RUN mkdir /data -p
RUN echo "Mes donnees" > /data/test
VOLUME /data

[root@centos1 monimage]# docker build -t monimage .
Sending build context to Docker daemon 2.048 kB
Step 1/6 : FROM centos:7
 ---> 8140d0c64310
Step 2/6 : RUN yum install -y wget
 ---> Running in 081f90b1d52c
Loaded plugins: fastestmirror, ovl
Determining fastest mirrors
 * base: centos.mirrors.ovh.net
…/...
 ---> Running in 3d4a557ff671
```

```
 ---> f11f404334b3
Removing intermediate container 3d4a557ff671
Successfully built f11f404334b3
```

```
[root@centos1 monimage]# docker run -d -it -P monimage
78d4ef6bc5395d1f302164fe08ce3478ec976ee398ebc973bd33f2a661cf396c

[root@centos1 monimage]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS              PORTS                                            NAMES
78d4ef6bc539        monimage            "/bin/bash"              8 seconds ago       Up 7
seconds        0.0.0.0:32773->80/tcp, 0.0.0.0:32772->8080/tcp   quizzical_kilby
aa8379b2e903        nginx               "nginx -g 'daemon ..."   5 minutes ago       Up 5
minutes        0.0.0.0:32771->80/tcp                            hardcore_jepsen

[root@centos1 monimage]# docker port 78d4ef6bc539
8080/tcp -> 0.0.0.0:32772
80/tcp -> 0.0.0.0:32773
```