

Version : **2022.01**

Dernière mise-à-jour : 2020/10/08 08:30

# DOF201 - Stocker les Images Docker

## Contenu du Module

- **DOF201 - Stocker les Images Docker**
  - Contenu du Module
  - LAB #1 - Installer un Registre Privé
    - 1.1 - Installer docker
    - 1.2 - Préparation
    - 1.3 - Créer un Registre local,
    - 1.4 - Créer un Serveur de Registre Dédié
      - Configurer le clone comme Registre Dédié
      - Configurer le Client

## LAB #1 - Installer un Registre Privé

### 1.1 - Installer docker

Docker n'est pas dans le dépôts de Debian. Afin de l'installer il convient d'ajouter le dépôt de docker. Premièrement, il est nécessaire d'installer les paquets permettant à Debian d'utiliser un dépôt en https :

```
root@debian9:~# apt-get update
...
root@debian9:~# apt-get install apt-transport-https ca-certificates curl gnupg2 software-properties-common
Reading package lists... Done
```

```
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version.
ca-certificates set to manually installed.
gnupg2 is already the newest version.
gnupg2 set to manually installed.
The following extra packages will be installed:
  libcurl3 python3-dbus python3-software-properties unattended-upgrades
Suggested packages:
  python-dbus-doc python3-dbus-dbg
The following NEW packages will be installed:
  apt-transport-https curl libcurl3 python3-dbus python3-software-properties
  software-properties-common unattended-upgrades
0 upgraded, 7 newly installed, 0 to remove and 1 not upgraded.
Need to get 960 kB of archives.
After this operation, 2,344 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Téléchargez la clef GPG officielle de docker :

```
root@debian9:~# curl -fsSL https://download.docker.com/linux/debian/gpg | apt-key add -
OK
```

Vérifiez que l'ID de la clef est **9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88** :

```
root@debian9:~# apt-key fingerprint 0EBFCD88
/etc/apt/trusted.gpg
-----
pub   4096R/0EBFCD88 2017-02-22
      Key fingerprint = 9DC8 5822 9FC7 DD38 854A  E2D8 8D81 803C 0EBF CD88
uid   Docker Release (CE deb) <docker@docker.com>
sub   4096R/F273FCD8 2017-02-22
...

```

Ajoutez le dépôt **stable** de docker :

```
root@debian9:~# add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb_release -cs) stable"
```



**Important** - Notez que la commande **lsb\_release -cs** retourne le nom de la distribution Debian, à savoir dans ce cas **stretch**.

Installez maintenant le paquet **docker-ce** :

```
root@debian9:~# apt-get update
...
root@debian9:~# apt-get install docker-ce
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  aufs-tools cgroupfs-mount git git-man libapparmor1 liberror-perl
  libnih-dbus1 libnih1 makedev mountall plymouth rsync
Suggested packages:
  git-daemon-run git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-arch git-cvs git-mediawiki git-svn plymouth-themes
The following NEW packages will be installed:
  aufs-tools cgroupfs-mount docker-ce git git-man libapparmor1 liberror-perl
  libnih-dbus1 libnih1 makedev mountall plymouth rsync
0 upgraded, 13 newly installed, 0 to remove and 99 not upgraded.
Need to get 26.5 MB of archives.
After this operation, 123 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Dernièrement, vérifiez la version de Docker client et serveur :

```
root@debian9:~# docker version
Client: Docker Engine - Community
 Version:           19.03.4
 API version:       1.40
 Go version:        go1.12.10
 Git commit:        9013bf583a
 Built:             Fri Oct 18 15:52:34 2019
 OS/Arch:           linux/amd64
 Experimental:      false

Server: Docker Engine - Community
 Engine:
  Version:           19.03.4
  API version:       1.40 (minimum version 1.12)
  Go version:        go1.12.10
  Git commit:        9013bf583a
  Built:             Fri Oct 18 15:51:05 2019
  OS/Arch:           linux/amd64
  Experimental:      false
containerd:
 Version:           1.2.10
 GitCommit:        b34a5c8af56e510852c35414db4c1f4fa6172339
runc:
 Version:           1.0.0-rc8+dev
 GitCommit:        3e425f80a8c931f88e6d94a8c831b9d5aa481657
docker-init:
 Version:           0.18.0
 GitCommit:        fec3683
```



**Important** - Docker est composé de trois éléments : un serveur, un client et un ou plusieurs **Repositories** ou Dépôts. Nous reviendrons sur les dépôts dans le détail plus tard dans ce cours.

Re-démarrez la machine virtuelle avant de poursuivre :

```
root@debian9:~# shutdown -r now
```

## 1.2 - Préparation

Vous devez vous connecter à la machine virtuelle debian9 à partir d'un terminal de votre gateway :

```
trainee@traineeXX:~$ ssh -l trainee 10.0.2.60
```

Passez en tant que **root** :

```
trainee@debian9:~$ su -  
Mot de passe : fenestros  
root@debian9:~#
```

Créez un répertoire nommé myDocker :

```
root@debian9:~# mkdir ~/myDocker  
root@debian9:~# cd ~/myDocker  
root@debian9:~/myDocker#
```

Créez le fichier myEntrypoint.sh :

```
root@debian9:~/myDocker# vi myEntrypoint.sh  
root@debian9:~/myDocker# cat myEntrypoint.sh  
#!/bin/bash  
if [ -z "$myVariable" ]; then  
    echo "La variable myVariable doit être renseignée"  
    return 1  
fi
```

```
while true;
do
    echo $1 \($(date +%H:%M:%S)\);
    sleep "$myVariable";
done
```

Testez ce script :

```
root@debian9:~/myDocker# myVariable=3 . ./myEntrypoint.sh salut
salut (20:04:39)
salut (20:04:42)
salut (20:04:45)
salut (20:04:48)
salut (20:04:51)
^C
root@debian9:~/myDocker#
```

Rendez ce script exécutable :

```
root@debian9:~/myDocker# chmod u+x myEntrypoint.sh
```

Créez maintenant le fichier **Dockerfile** dans le répertoire **~/myDocker** :

```
root@debian9:~/myDocker# vi Dockerfile
root@debian9:~/myDocker# cat Dockerfile
FROM centos:latest
MAINTAINER i2tch "infos@i2tch.eu"
COPY myEntrypoint.sh /entrypoint.sh
ENV myVariable 3
ENTRYPOINT ["/entrypoint.sh"]
CMD ["mycommand"]
```

Générez maintenant l'image :

```
root@debian9:~/myDocker# docker build -t i2tch/mydocker .
Sending build context to Docker daemon 3.072kB
Step 1/6 : FROM centos:latest
----> 9f38484d220f
Step 2/6 : MAINTAINER i2tch "infos@i2tch.eu"
----> Running in 02c700ed04da
Removing intermediate container 02c700ed04da
----> 4274107d52e2
Step 3/6 : COPY myEntrypoint.sh /entrypoint.sh
----> 7a3923372768
Step 4/6 : ENV myVariable 3
----> Running in 3288bf6291ad
Removing intermediate container 3288bf6291ad
----> 3edb630c1511
Step 5/6 : ENTRYPOINT ["/entrypoint.sh"]
----> Running in 8dcba2c41520
Removing intermediate container 8dcba2c41520
----> 11962052539c
Step 6/6 : CMD ["mycommand"]
----> Running in f891fbcfaad0
Removing intermediate container f891fbcfaad0
----> 7925ba23abb2
Successfully built 7925ba23abb2
Successfully tagged i2tch/mydocker:latest
```

### 1.3 - Installer un Registre Local

Pour installer un registre privé, il convient d'utiliser une image publique de docker :

```
root@debian9:~/bestp# cd ..
root@debian9:~# docker run -d --name registry -p 88:5000 registry:2.0
Unable to find image 'registry:2.0' locally
```

```
2.0: Pulling from library/registry
4d2e9ae40c41: Pull complete
a3ed95caeb02: Pull complete
7c8152785df5: Pull complete
8b04aafd7cd8: Pull complete
c97c75d2d42e: Pull complete
4b3ef98bba76: Pull complete
edee0288d356: Pull complete
ea2a9399d365: Pull complete
ddf532273b60: Pull complete
e9e91aa1843e: Pull complete
6144b0ffbb4c: Pull complete
Digest: sha256:3cac1869696e4ff3435bbc30391749ac373f7471736dbb48dfa9bfde08c4efd2
Status: Downloaded newer image for registry:2.0
c4c7cad999cdd77df78a21897bd11c4742a094c1c76a2134fbe4a4d2d92bff0e
```

Utilisez maintenant **lynx** à partir d'un terminal de votre machine **hôte Docker** pour vérifier que le registre est actif :

```
root@debian9:~# lynx --dump http://localhost:88/v2
{}root@debian9:~#
```



**Important** - Notez la réponse du serveur est **{}** soit une liste JSON vide.

Renommez l'image **i2tch/mydocker** afin de pointer vers le nouveau registre :

```
root@debian9:~# docker tag i2tch/mydocker localhost:88/mydocker
```



**Important** - Notez que le tag **i2tch/mydocker** cache le nom du registre par défaut qui est le registre public de Docker Hub.

Envoyez votre image **localhost:88/mydocker** sur ce nouveau registre :

```
root@debian9:~# docker push localhost:88/mydocker
The push refers to a repository [localhost:88/mydocker]
873a8ac77d4d: Pushed
b362758f4793: Pushed
latest: digest: sha256:30866da81d92d2a1015b869c596ddd6e188f33894c41d8effa2161e5c2862b1f size: 5531
```

Constatez maintenant la présence de l'image dans le registre :

```
root@debian9:~# lynx --dump http://localhost:88/v2/mydocker/tags/list
{"name": "mydocker", "tags": ["latest"]}
```

```
root@debian9:~# docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
i2tch/mydocker      latest      c37edbd43993     45 minutes ago  193MB
localhost:88/mydocker latest      c37edbd43993     45 minutes ago  193MB
registry            2.0        3bccd459597f     2 years ago     549MB
```

## 1.4 - Créer un Serveur de Registre Dédié

Actuellement, le registre privé créé ci-dessus n'est pas accessible à partir du réseau local car il est référencé par localhost. Il convient donc maintenant de mettre en place un serveur dédié.

Connectez-vous à la VM Debian\_9\_1 :

```
trainee@traineeXX:~$ ssh -l trainee 10.0.2.61
```

En tant que root, modifiez le nom d'hôte de la machine :

```
trainee@debian91:~$ su -
Mot de passe : fenestros
```

```
root@debian91:~# nmcli general hostname myregistry.i2tch.loc
root@debian91:~# hostname
myregistry.i2tch.loc
```

Installez Docker :

```
root@debian91:~# apt-get update
root@debian91:~# apt-get install apt-transport-https ca-certificates curl gnupg2 software-properties-common
root@debian91:~# curl -fsSL https://download.docker.com/linux/debian/gpg | apt-key add -
root@debian91:~# add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb_release -cs)
stable"
root@debian91:~# apt-get update
root@debian91:~# apt-get install docker-ce
```

Editez le fichier **/etc/hosts** et changez l'entrée pour l'adresse IP 10.0.2.61 :

```
root@debian91:~# vi /etc/hosts
root@debian91:~# cat /etc/hosts
127.0.0.1    localhost
10.0.2.60   debian9.i2tch.loc    debian9
10.0.2.61   myregistry.i2tch.loc    myregistry
10.0.2.62   manager.i2tch.loc     manager
10.0.2.63   worker1.i2tch.loc     worker1
10.0.2.64   worker2.i2tch.loc     worker2

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

Créez maintenant un certificat auto-signé avec **openssl** :

```
root@debian91:~# cd / && mkdir certs && openssl req -newkey rsa:4096 -nodes -sha256 -keyout certs/domain.key -
x509 -days 365 -out certs/domain.crt
```

```
Generating a 4096 bit RSA private key
```

```
.....++
```

```
.....
```

```
.....++
```

```
writing new private key to 'certs/domain.key'
```

```
-----
```

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

```
-----
```

```
Country Name (2 letter code) [AU]:FR
```

```
State or Province Name (full name) [Some-State]:VAR
```

```
Locality Name (eg, city) []:Toulon
```

```
Organization Name (eg, company) [Internet Widgits Pty Ltd]:I2TCH LTD
```

```
Organizational Unit Name (eg, section) []:TRAINING
```

```
Common Name (e.g. server FQDN or YOUR name) []:myregistry
```

```
Email Address []:
```

```
root@registry:/# ls certs/
```

```
domain.crt  domain.key
```

Créez un conteneur en mode sécurisé avec TLS à partir de l'image registry :

```
root@registry:/# docker run -d -p 5000:5000 --name registry -v `pwd`/certs:/certs -e
REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt -e REGISTRY_HTTP_TLS_KEY=/certs/domain.key registry:2.0
943c01b67cf3f461270a55ac3d9df6622cc9d74e5f272e17153183ff29ee5932
root@debian9:/#
```

```
root@registry:/# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
943c01b67cf3	registry:2.0	"registry cmd/regi..."	31 seconds ago	Up 24 seconds

```
0.0.0.0:5000->5000/tcp registry
```

Envoyez une copie du fichier **/certs/domain.crt** vers le répertoire /tmp de la machine virtuelle **Debian\_9** en le renommant **ca.crt** :

```
root@debian91:~# scp /certs/domain.crt trainee@10.0.2.60:/tmp/ca.crt
The authenticity of host '10.0.2.60 (10.0.2.60)' can't be established.
ECDSA key fingerprint is 79:00:60:0e:2b:71:5e:cb:1a:08:45:e8:ab:45:b8:dd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.60' (ECDSA) to the list of known hosts.
trainee@10.0.2.60's password: trainee
domain.crt
100% 2017      2.0KB/s   00:00
```

## Configurer le Client

Sortez de la VM **debian91** et connectez-vous à la VM **debian9** :

```
trainee@traineeXX:~$ ssh -l trainee 10.0.2.60
```

Passez en tant que **root** :

```
trainee@debian9:~$ su -
Mot de passe : fenestros
root@debian9:~#
```

Supprimez le conteneur **registry** :

```
root@debian9:~# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
c4c7cad999cd	registry:2.0	"registry cmd/regi..."	4 hours ago	Exited (2) 4 hours ago
registry				

```
21b0490a93dd      i2tch/mydocker    "/entrypoint.sh my..." 4 hours ago      Exited (137) 4 hours ago
myDocker

root@debian9:~# docker rm registry
registry

root@debian9:~# docker ps -a
CONTAINER ID      IMAGE              COMMAND              CREATED           STATUS
PORTS            NAMES
21b0490a93dd      i2tch/mydocker    "/entrypoint.sh my..." 4 hours ago      Exited (137) 4 hours ago
myDocker
```

ainsi que l'image du registry :

```
root@debian9:~# docker rmi registry:2.0
Untagged: registry:2.0
Untagged: registry@sha256:3cac1869696e4ff3435bbc30391749ac373f7471736dbb48dfa9bfde08c4efd2
Deleted: sha256:3bccd459597f38e78ce95a408e506099644ca713d79157d2f3e3a7975f1c9146
Deleted: sha256:e486d081a2821af119adbc458214cacf510ebb519cdf31e04ac3d4c6ac521c31
Deleted: sha256:d206757aba612d90965729f186dfb70aa5e63b53dafd7e7b20a76005acddfe24
Deleted: sha256:1b953394add763fbaa50446b03d636f453fa75b2d12676ceb4b71681b4da6e7d
Deleted: sha256:e8a2cacb793cb843e35bbf64e9c73d685d83ecc8192fbdbc098b0e5fbcced848
Deleted: sha256:19eec9bf8dcc3cb15b93cdcff717057444c540ee641cf20b7ef647d19b9d1df4
Deleted: sha256:2fb6f2d834e4dd55a7a6bb1472f5c293242e6439982332143d7332642a238d40
Deleted: sha256:e08b039d73e73f17b3fa02ebf9a28ef156a7b43b828cd67ed422b9d7a01e5482
Deleted: sha256:bd85e070a996530cc5f576bc5f56a0e3a159e00091410987416844de6e018415
Deleted: sha256:6269effe5aa88dfc0071a2841605d95c0d809111333b5044b2459b43e5879f44
Deleted: sha256:784ff50f126370cb3be928cd14a078605990efda458e73ad43f2a2e40e1efda2
Deleted: sha256:29efd197b6056e252ebb46fe1b315489c2f8e032eb32d1f6fcd0119ee9536df7
Deleted: sha256:3d42e373d713bfab9403159f282ba4cfaa7b4f33e57d5e36d4d2ca0ca1a2f4e7
Deleted: sha256:518c6772b2fc316c63e9f4f9745e3587f169ec916fd26749b0ce7bf1f36bb93b
Deleted: sha256:e74bff63859c2597691fffd6fe95b35b0803f6791bc6565072bf07067574dba3
Deleted: sha256:f76baad394e6d835fa0e166254e9f70af873fe43232c67ac198a67865084ccee
Deleted: sha256:b0eba742532f664bab6fc6d5bd00c60141397ba299f6743457f7ead64d7c714d
```

```
Deleted: sha256:c10c49233bc62ce69aafa1a44a62d8639a7be08d68b1c23d6b25ebb13e5324b0
Deleted: sha256:1c8e9ba14469736fa03ede127e47f5821ca97b3029385ec1348e87932c875ff8
Deleted: sha256:611f5c9e21fce8d870f9ba82248980ca3baaaf95afef2b17504372d1ca03cb5a
Deleted: sha256:4cb1abe123254326dba72305ef897fdf5364ebb1823413e13dbced41990f951e
Deleted: sha256:e10e5ea91f007db418b284f4adc5f0b98f374d79ae52b9687b0d6d33865ffbcf
Deleted: sha256:c69ae1aa46985cbaf186b6354c61a1d2e0d6af47133db47bf04f0c6eb9c858e9
```

```
root@debian9:~# docker images
REPOSITORY          TAG                IMAGE ID           CREATED           SIZE
i2tch/mydocker      latest            c37edbd43993      5 hours ago      193MB
localhost:88/mydocker latest            c37edbd43993      5 hours ago      193MB
```

Renommez l'image **i2tch/mydocker** afin de pointer vers le serveur de registre :

```
root@debian9:~# docker tag i2tch/mydocker myregistry:5000/mydocker
root@debian9:~# docker images
REPOSITORY          TAG                IMAGE ID           CREATED           SIZE
i2tch/mydocker      latest            c37edbd43993      5 hours ago      193MB
localhost:88/mydocker latest            c37edbd43993      5 hours ago      193MB
myregistry:5000/mydocker latest            c37edbd43993      5 hours ago      193MB
```

Editez le fichier **/etc/hosts** afin de pointer le 10.0.2.61 vers le nom **myregistry** :

```
root@debian9:~# vi /etc/hosts
root@debian9:~# cat /etc/hosts
127.0.0.1    localhost
10.0.2.60   debian9.i2tch.loc    debian9
10.0.2.61   myregistry.i2tch.loc myregistry
10.0.2.62   manager.i2tch.loc    manager
10.0.2.63   worker1.i2tch.loc    worker1
10.0.2.64   worker2.i2tch.loc    worker2

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
```

```
ff02::1 ip6-allnodes  
ff02::2 ip6-allrouters
```

Déplacez le fichier **/tmp/ca.crt** vers le répertoire **/etc/docker/certs.d/myregistry:5000/** :

```
root@debian9:~# mkdir -p /etc/docker/certs.d/myregistry:5000  
root@debian9:~# mv /tmp/ca.crt /etc/docker/certs.d/myregistry:5000/
```

Testez la réponse du registre :

```
root@debian9:~# curl http://myregistry:5000/v2/  
□□□□
```

Finalement, envoyez l'image au registre :

```
root@debian9:~# docker push myregistry:5000/mydocker  
The push refers to a repository [myregistry:5000/mydocker]  
873a8ac77d4d: Pushed  
b362758f4793: Pushed  
latest: digest: sha256:30866da81d92d2a1015b869c596ddd6e188f33894c41d8effa2161e5c2862b1f size: 5531
```