

# Doc Job09

---

created : 24-06-02 21:31

modified : 24-06-02 21:31

tags : [note]

References :

---

<https://sysdig.com/learn-cloud-native/container-security/what-is-a-docker-registry/>

<https://devopssec.fr/article/deployer-manipuler-securiser-un-serveur-registry-docker-prive>

Un registre Docker local est un dépôt privé où vous pouvez stocker et distribuer vos images Docker. Il fonctionne de manière similaire au Docker Hub, mais il est hébergé sur votre propre infrastructure. Cela permet un meilleur contrôle des images et une meilleure sécurité, car les images ne quittent jamais votre réseau local.

Doc tirée de ce [tuto](#) :

```
sudo docker pull (nom d'une image qui est installée sur mon pc ex: nginx)
```

```
sudo docker run -d -p 5000:5000 --name dockerregistry registry:latest
```

On vérifie si l'image s'est run:

```
docker ps
```

Je vérifie que mon registry est up:

```
sudo curl -X GET http://192.168.70.132:5000/v2/_catalog
```

```
sudo docker run -d -p 5000:5000 --restart=always --name  
registry -v $(pwd)/docker-registry:/var/lib/registry  
registry:latest
```

(rm si un container registry tourne déjà)

Ca a créé un repository, on vérifie qu'il existe

Nous allons maintenant push une image docker (celle de nginx que nous avons créé au début par exemple)

```
sudo docker tag nginx 192.168.70.132:5000/nginx
```

On vérifie que l'image s'est bien créée :

```
[root@node1 dockerregistry]# docker tag nginx 192.168.145.130:5000/nginx  
[root@node1 dockerregistry]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
192.168.145.130:5000/nginx	latest	822b7ec2aaf2	2 days ago	133MB
nginx	latest	822b7ec2aaf2	2 days ago	133MB
registry	latest	b2cb11db9d3d	4 days ago	26.2MB
thetips4you/nodeapp_test	latest	5d7c407732e2	3 weeks ago	915MB
node	latest	1d88d89c7230	4 weeks ago	907MB
docker.elastic.co/kibana/kibana	7.13.4	9ebf2fabb05d	7 weeks ago	1.35GB
docker.elastic.co/elasticsearch/elasticsearch	7.13.4	1c9757417a29	7 weeks ago	1.02GB
grafana/grafana	latest	38209d2ec29a	2 months ago	206MB
jenkins/jenkins	latest	ab631cb41bd5	2 months ago	569MB
docker.bintray.io/jfrog/artifactory-cpp-ce	latest	48da60366a1f	3 months ago	790MB
docker.elastic.co/elasticsearch/elasticsearch	7.12.1	41dc8ea0f139	4 months ago	851MB
alpine	latest	6dbb9cc54074	4 months ago	5.61MB

```
sudo docker push 192.168.70.132:5000/nginx
```

On va configurer le daemon :

```
sudo nano /etc/docker/daemon.json
```

On y entre ce paramètre:

```
{  
  "insecure-registries":["192.168.70.132:5000"]  
}
```

On redémarre le daemon docker:

```
sudo systemctl stop docker
```

```
sudo systemctl start docker
```

On vérifie si docker s'est bien relancé :

```
sudo systemctl status docker
```

Ensuite :

```
sudo docker push 192.168.70.132:5000/nginx
```

```
sudo docker pull 192.168.70.132:5000/nginx
```

Et puis on se connecte en http:

```
http://192.168.70.132:5000/v2/_catalog
```

