Docker Registry

Exercice 1 : Registry privé local de base pour le principe

Exécuter la commande suivante :

```
# docker run -d -p 5000:5000 --restart=always --name registry \
                    -v /mnt/registry:/var/lib/registry \
    registry:2
```

Consulter l'url : http://localhost:5000/v2

Downloader l'image busybox.

De l'image busybox, créer un tag nommé 'localhost:5000/busybox:1.0'.

Vérifier en listant les images.

Transférer l'image au registry.

Vérifier avec :

http://localhost:5000/v2/_catalog
http://localhost:5000/v2/busybox/tags/list

Supprimer du poste l'image ' localhost:5000/busybox:1.0', et vérifier que l'image soit bien supprimée.

Transférer l'image busybox du registry sur le poste hôte, ou exécuter un conteneur de cette image.

Exercice 2 : Registry privé avec un accés dans un environnement de confiance (insecure registries)

Exécuter la commande 'docker info' pour consulter la section 'Insecure Registries'.

Mettre à jour le fichier /etc/docker/daemon.json (ou le créer) pour autoriser l'utilisation de notre registry en mode non sécurisé. L'accés sera possible par d'autres postes sans l'utilisation de clés, ce qui est envisageable au sein d'un réseau de confiance.

```
# vi /etc/docker/daemon.json
{ "insecure-registries": ["192.168.1.12:5000"] }
          Utiliser l'adresse IP du host du registry
```

Exécuter la commande 'docker info' pour consulter la section 'Insecure Registries'.

Downloader l'image busybox de l'exercice précédent par la commande suivante, puis vérifier :

```
# docker pull 192.168.1.12:5000/busybox:1.0
# docker images
```

Downloader de Docker Hub l'image hello-world. Créer un tag et transférer cette image au sein du registry (en utilisant l'adresse IP et non localhost).

Vérifier avec :

http://localhost:5000/v2/_catalog
http://192.168.1.12:5000/v2/hello-world/tags/list

Nettoyage :

```
# mv  /etc/docker/daemon.json  /etc/docker/old.daemon.json.old
# docker rmi 192.168.1.12:5000/busybox:1.0
```

Vérification :

https://192.168.1.12/v2/busybox/tags/list

Exercice 3 : Registry privé sécurisé et avec authentification

Au cas où, installer la commande htpasswd via le package httpd-tools.

```
# yum install -y httpd-tools
```

Création d'un compte pour l'authentification:

Récupérer l'image 'registry:2'.

Gestion d'un compte pour l'authentification :

Syntaxe de la commande :

docker run --entrypoint htpasswd registry:2 -Bbn votrelogin votrepassword > /auth/htpasswd

Créer le répertoire /opt/jmb/auth, puis utiliser la syntaxe de la commande présentée ci-dessus pour créer un utilisateur admin (mot de passe admin1234) pour l'authentification.

```
# mkdir   -p   /opt/jmb/auth
# docker   run   --entrypoint   htpasswd   registry:2  \
          -Bbn   admin   admin1234   >   /opt/jmb/auth/htpasswd
```

Création du certificat :

Créer le répertoire /opt/jmb/auth.

Exécuter la commande suivante pour la création du certificat :

Remarque: pour le 'Common Name', vous indiquerez le hostname de votre poste.

```
# openssl   req   -newkey  rsa:4096   -nodes   -sha256   \
     -keyout   /opt/jmb/certs/domain.key   -x509   -days  365   \
     -out   /opt/jmb/certs/domain.crt
. . .
Country Name (2 letter code) [XX]:FR
State or Province Name (full name) []:paris
Locality Name (eg, city) [Default City]:paris
Organization Name (eg, company) [Default Company Ltd]:spherius
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:centos
Email Address []:
```

Verifier si 2 fichiers ont été créé au sein du répertoire.

Gestion pour le client :

Créer le répertoire '/etc/docker/certs.d/centos:5000' et y copier le fichier /opt/jmb/certs/domain.crt.

Exécuter les commandes suivantes pour prendre en comptes ces nouveaux paramètres.

```
# systemctl   daemon-reload
# systemctl   restart   docker
```

Démarrer un registry via la commande suivante.

```
# docker   run   -d   -p  5000:5000   --restart=always   --name
registry   \
     -v   /opt/jmb/registry:/var/lib/registry   \
     -v   /opt/jmb/certs:/certs   \
          -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt   \
          -e REGISTRY_HTTP_TLS_KEY=/certs/domain.key   \
     -v /opt/jmb/auth:/auth   \
          -e REGISTRY_AUTH=htpasswd   \
          -e REGISTRY_AUTH_HTPASSWD_REALM="Registry Realm"   \
          -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd   \
     registry:2
```

S'authentifier au registry :

```
# docker  login  centos:5000
```

D'une image de votre choix, créer un tag et copier l'image au sein du registry.

```
# docker  images
REPOSITORY         TAG                IMAGE ID            CREATED             SIZE
img1               latest             43f0d7fef7a3        6 months ago        1.95kB

# docker   tag   img1   centos:5000/img1

# docker   push   centos:5000/img1
```

Se déconnecter :

```
# docker   logout   centos:5000
```

Sur un autre poste :

Mettre à jour le fichier /etc/hosts pour ajouter une ligne concernant le poste où fonctionne le registry.

```
[client]# vi   /etc/hosts
192.168.1.4    centos      # @IP_poste_registry   hostname_poste_registry
```

Gestion pour le client (comme précédemment) :

```
[client]# mkdir   -p    /etc/docker/certs.d/centos:5000
[client]# scp   192.168.1.4:/opt/jmb/certs/domain.crt  \
                /etc/docker/certs.d/centos:5000
[client]# ls /etc/docker/certs.d/centos:5000
domain.crt
```

S'authentifier au registry.

```
[client]# docker   login   centos:5000
```

Récupérer l'image sur le poste client.

```
[client]# docker   pull   centos:5000/img1
```

Vérifier que l'image est disponible sur le poste client.

```
[client]# docker   images
```

Se déconnecter du registry.

```
[client]# docker   logout   centos:5000
```

On peut consulter le registry avec la commande curl :

[client]# curl --insecure -u "admin:admin1234" https://centos/v2/
[client]# curl –insecure -u "admin:admin1234" https://centos/v2/_catalog
[client]# curl –insecure -u "admin:admin1234" https://centos/v2/busybox/tags/list

ou via le browser : https://centos/v2/_catalog (demande d'authentification)
https://centos/v2/busybox/tags/list

Nettoyage :

[client]# docker rmi centos:5000/img1

Sur le poste du registry :

docker rm -f registry
docker rmi centos:5000/img1 registry
rm -rf /opt/jmb/registry

Correction – Docker Registry

Exercice 1 : Registry privé local de base pour le principe

Exécuter la commande suivante :

```
# docker run -d -p 5000:5000 --restart=always --name registry \
                    -v /mnt/registry:/var/lib/registry \
    registry:2
```

Consulter l'url : http://localhost:5000/v2

Downloader l'image busybox.

```
# docker pull busybox
```

De l'image busybox, créer un tag nommé 'localhost:5000/busybox:1.0'.

```
# docker tag busybox localhost:5000/busybox:1.0
```

Vérifier en listant les images.

```
# docker images
```

Transférer l'image au registry.

```
# docker push localhost:5000/busybox:1.0
```

Vérifier avec :

http://localhost:5000/v2/_catalog
http://localhost:5000/v2/busybox/tags/list

Supprimer du poste l'image ' localhost:5000/busybox:1.0', et vérifier que l'image soit bien supprimée.

```
# docker rmi  localhost:5000/busybox:1.0
# docker images
```

Transférer l'image busybox du registry sur le poste hôte, ou exécuter un conteneur de cette image.

```
# docker pull localhost:5000/busybox:1.0
# docker images

ou    # docker  run  -dit  --name cont1  localhost:5000/busybox:1.0
      ; exit
      # docker ps -a        ;     docker images         ;     docker rm cont1
```

Exercice 2 : Registry privé avec un accés dans un environnement de confiance (insecure registries)

Exécuter la commande 'docker info' pour consulter la section 'Insecure Registries'.

```
# docker info
. . .
Insecure Registries:
 127.0.0.0/8
. . .
```

Mettre à jour le fichier /etc/docker/daemon.json (ou le créer) pour autoriser l'utilisation de notre registry en mode non sécurisé. L'accés sera possible par d'autres postes sans l'utilisation de clés, ce qui est envisageable au sein d'un réseau de confiance.

```
# vi /etc/docker/daemon.json
{ "insecure-registries": ["192.168.1.12:5000"] }
          Utiliser l'adresse IP du host du registry
```

Exécuter la commande 'docker info' pour consulter la section 'Insecure Registries'.

```
# docker info
. . .
Insecure Registries:
 192.168.1.12:5000
 127.0.0.0/8
. . .
```

Downloader l'image busybox de l'exercice précédent par la commande suivante, puis vérifier :

```
# docker pull 192.168.1.12:5000/busybox:1.0
# docker images
```

Downloader de Docker Hub l'image hello-world. Créer un tag et transférer cette image au sein du registry (en utilisant l'adresse IP et non localhost).

```
# docker pull hello-world
# docker tag hello-world 192.168.1.12:5000/hello-world:2.0
# docker push 192.168.1.12:5000/hello-world:2.0
```

Vérifier avec :

http://localhost:5000/v2/_catalog
http://192.168.1.12:5000/v2/hello-world/tags/list

Nettoyage :

```
# mv  /etc/docker/daemon.json  /etc/docker/old.daemon.json.old
# docker rmi 192.168.1.12:5000/busybox:1.0
```

Vérification :

https://192.168.1.12/v2/busybox/tags/list

Exercice 3 : Registry privé sécurisé et avec authentification

Au cas où, installer la commande htpasswd via le package httpd-tools.

```
# yum install -y httpd-tools
```

Création d'un compte pour l'authentification:

Récupérer l'image 'registry:2'.

```
# docker pull registry:2
```

Gestion d'un compte pour l'authentification :

Syntaxe de la commande :

docker run --entrypoint htpasswd registry:2 -Bbn votrelogin votrepassword > /auth/htpasswd

Créer le répertoire /opt/jmb/auth, puis utiliser la syntaxe de la commande présentée ci-dessus pour créer un utilisateur admin (mot de passe admin1234) pour l'authentification.

```
# mkdir   -p   /opt/jmb/auth
# docker   run   --entrypoint   htpasswd   registry:2  \
          -Bbn   admin   admin1234   >   /opt/jmb/auth/htpasswd
```

Création du certificat :

Créer le répertoire /opt/jmb/auth.

```
# mkdir   -p   /opt/jmb/certs
# cd   /opt/jmb/certs
```

Exécuter la commande suivante pour la création du certificat :

Remarque: pour le 'Common Name', vous indiquerez le hostname de votre poste.

```
# openssl   req   -newkey  rsa:4096   -nodes   -sha256   \
     -keyout   /opt/jmb/certs/domain.key   -x509   -days  365   \
     -out   /opt/jmb/certs/domain.crt
Generating a 4096 bit RSA private key
.........................++
.........................++
writing new private key to '/opt/jmb/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) [XX]:FR
State or Province Name (full name) []:paris
Locality Name (eg, city) [Default City]:paris
```

```
Organization Name (eg, company) [Default Company Ltd]:spherius
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:centos
Email Address []:
```

Verifier si 2 fichiers ont été créé au sein du répertoire.

```
# ls   /opt/jmb/certs
```

Gestion pour le client :
Créer le répertoire '/etc/docker/certs.d/centos:5000' et y copier le fichier /opt/jmb/certs/domain.crt.

```
# mkdir  -p  /etc/docker/certs.d/centos:5000/
# cp   /opt/jmb/certs/domain.crt   /etc/docker/certs.d/centos:5000/
# ls   /etc/docker/certs.d/centos:5000/
domain.crt
```

Exécuter les commandes suivantes pour prendre en comptes ces nouveaux paramètres.

```
# systemctl   daemon-reload
# systemctl   restart   docker
```

Démarrer un registry via la commande suivante.

```
# docker   run   -d   -p  5000:5000   --restart=always   --name
registry   \
     -v   /opt/jmb/registry:/var/lib/registry   \
     -v   /opt/jmb/certs:/certs   \
          -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt   \
          -e REGISTRY_HTTP_TLS_KEY=/certs/domain.key   \
     -v /opt/jmb/auth:/auth   \
          -e REGISTRY_AUTH=htpasswd   \
          -e REGISTRY_AUTH_HTPASSWD_REALM="Registry Realm"   \
          -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd   \
     registry:2
```

S'authentifier au registry :

```
# docker  login  centos:5000
Username: admin
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

D'une image de votre choix, créer un tag et copier l'image au sein du registry.

```
# docker  images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
img1                latest              43f0d7fef7a3        6 months ago        1.95kB

# docker   tag   img1   centos:5000/img1

# docker   push   centos:5000/img1
The push refers to repository [centos:5000/img1]
a310e7bb9dce: Pushed
latest: digest:
sha256:13613b4461cb1e4e83d96977808a3d4a6001900f089c62549a18e194d88e32c9 size: 524
```

Se déconnecter :

```
# docker   logout   centos:5000
```

Sur un autre poste :

Mettre à jour le fichier /etc/hosts pour ajouter une ligne concernant le poste où fonctionne le registry.

```
[client]# vi   /etc/hosts
192.168.1.4    centos      # @IP_poste_registry   hostname_poste_registry
```

Gestion pour le client (comme précédemment) :

```
[client]# mkdir   -p    /etc/docker/certs.d/centos:5000
[client]# scp   192.168.1.4:/opt/jmb/certs/domain.crt  \
                /etc/docker/certs.d/centos:5000
[client]# ls /etc/docker/certs.d/centos:5000
domain.crt
```

S'authentifier au registry.

```
[client]# docker   login   centos:5000
Username: admin
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Récupérer l'image sur le poste client.

```
[client]# docker   pull   centos:5000/img1
Using default tag: latest
latest: Pulling from img1
97bdff37536f: Pull complete
Digest: sha256:13613b4461cb1e4e83d96977808a3d4a6001900f089c62549a18e194d88e32c9
Status: Downloaded newer image for centos:5000/img1:latest
```

Vérifier que l'image est disponible sur le poste client.

```
[client]# docker   images
REPOSITORY            TAG                 IMAGE ID            CREATED             SIZE
centos:5000/img1      latest              43f0d7fef7a3        6 months ago        1.95kB
```

Se déconnecter du registry.

```
[client]# docker   logout   centos:5000
```

On peut consulter le registry avec la commande curl :

[client]# curl --insecure -u "admin:admin1234" https://centos/v2/
[client]# curl –insecure -u "admin:admin1234" https://centos/v2/_catalog
[client]# curl –insecure -u "admin:admin1234" https://centos/v2/busybox/tags/list

ou via le browser : https://centos/v2/_catalog (demande d'authentification)
https://centos/v2/busybox/tags/list

Nettoyage :

[client]# docker rmi centos:5000/img1

Sur le poste du registry :

```
# docker   rm   -f   registry
# docker   rmi   centos:5000/img1    registry
# rm   -rf   /opt/jmb/registry
```