# **Tests avec docker-compose.yml :**

## Présentation :

3 traitements ont été créés afin de tester l'accès aux requêtes des 3 Endpoint /info, /predict et /score

### Test /info

Le traitement “tests/repo\_test\_acces\_info/**test\_access\_info.py**” permet de vérifier l’accès aux informations selon les cas suivants :

* Utilisation d’une authentification valide en format base64 + user habilité. Le code statut attendu : 200
* Utilisation d’un mauvais format. Le code statut attendu : 400
* Utilisation d’un format en base64 mais d’un user non habilité. Le code statut : 403

### Test /predict

Le traitement “tests/repo\_test\_acces\_predict/**test\_access\_predict.py**” permet de vérifier l’accès aux prédictions du modèle SVC selon les cas suivants :

* En utilisant les données du fichier “test.csv” avec un format “csv”. Le code statut attendu : 200
* En utilisant les données du fichier “test.csv” avec un format autre que “csv”. Le code statut attendu : 400

### Test /score

Le traitement “tests/repo\_test\_acces\_score/**test\_access\_score.py**” permet de vérifier l'accès au score du modèle SVC. Le code statut attendu : 200

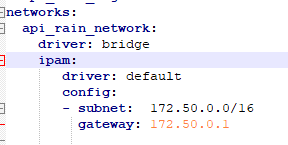
Les 3 traitements vont écrire le résultat des tests dans le fichier “log.txt” accessible dans le répertoire du volume **api\_rain\_log** dans : '/var/lib/docker/volumes/tests\_api\_rain\_log/\_data'

## Lancement:

### Fichier “docker-compose.yml”

Le lancement des 3 tests se fait en utilisant le fichier “tests/docker-compose.yml”. Ce fichier permet la création et le lancement des containers pour chacun des 3 tests en plus du container de l’API à interroger.

Les containers créés partagent le même NETWORK : api\_rain\_network



Et les 3 containers des tests auront comme volume partagé api\_rain\_log

### Script “tests/setup.sh”

Afin d’exécuter les 3 tests, il faut d’abord lancer le traitement “tests/**setup.sh**” qui permet de :

* Construire les différentes images Docker pour :
  + Les 3 tests via les fichiers Dockerfile présents dans chacun des répertoires où se trouvent les traitements.
  + L’API à interroger via le fichier Dockerfile présent dans le répertoire “mai21\_cde\_rains”
* Exécuter le fichier “yml” via la commande “docker-compose up”

### Script “tests/off.sh”

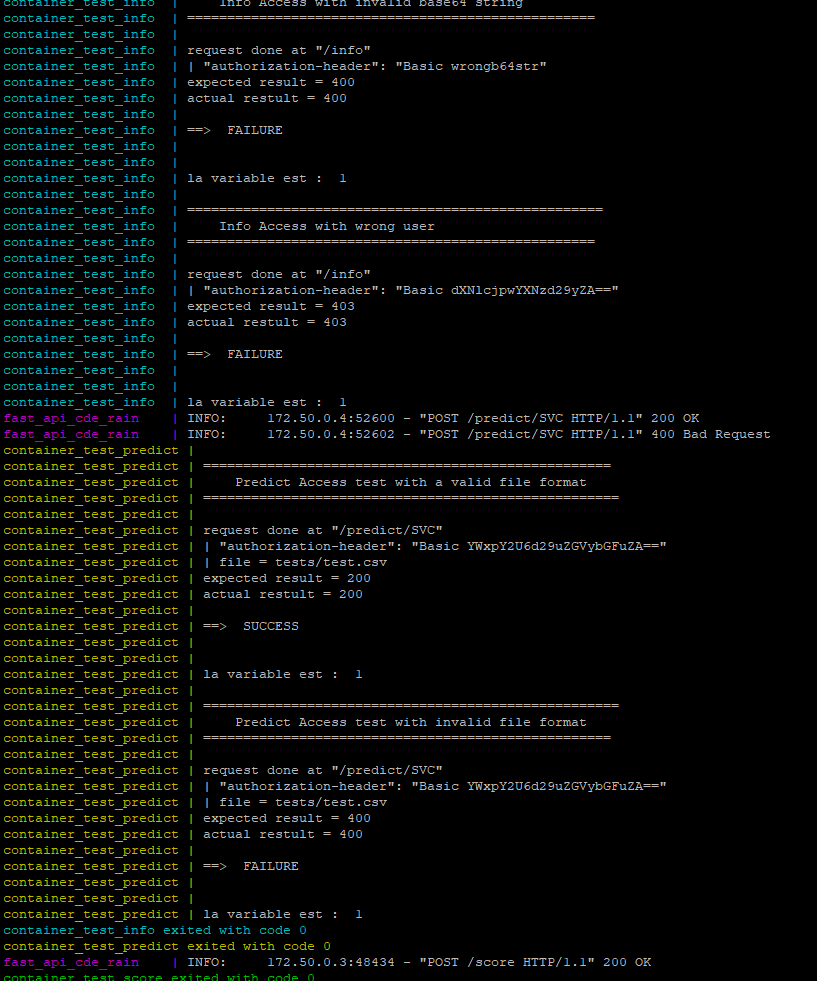
Le traitement “tests/**off.sh**” permet la suppression l’intégralité des containers et des images créées dans le cadre des tests.

## Démonstration:

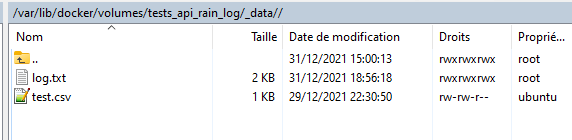
1. Lancement du traitement “setup.sh”



Le détail de création des images + la log générée par les tests vont apparaître dans le terminal :



1. Vérifier l’alimentation du fichier “log.txt” à l’emplacement du volume “api\_rain\_log” :



Aperçu du contenu du fichier :

