

TP 1:Initialisation a Hadoop

Partie 1 : Installation et Configuration

Étape 1 : Clonage du référentiel Hadoop

La première étape consiste à cloner le dépôt du cluster Hadoop.

Étape 2 : Accéder au Dossier Cloné

Une fois le dépôt cloné, entrez dans le dossier contenant le cluster Hadoop en utilisant la commande :

```
9/27/20 C:\Users\nourh>cd C:\Users\nourh\docker-hadoop
```

<div> <div> <div>↻</div> <div>🖥️</div> </div> <div> <div>></div> <div>This PC</div> <div>></div> <div>OS (C:)</div> <div>></div> <div>Users</div> <div>></div> <div>nourh</div> <div>></div> <div>docker-hadoop</div> <div>></div> </div> </div>				
<div> <div>📁</div> <div>📄</div> <div>📄</div> <div>🔗</div> <div>🗑️</div> <div>⬆️ Sort ▾</div> <div>☰ View ▾</div> <div>⋮</div> </div>				
Name	Date modified	Type	Size	
📁 base	9/27/2024 2:33 PM	File folder		
📁 datanode	9/27/2024 2:33 PM	File folder		
📁 historyserver	9/27/2024 2:33 PM	File folder		
📁 namenode	9/27/2024 2:33 PM	File folder		
📁 nginx	9/27/2024 2:33 PM	File folder		
📁 nodemanager	9/27/2024 2:33 PM	File folder		
📁 resourcemanager	9/27/2024 2:33 PM	File folder		
📁 submit	9/27/2024 2:33 PM	File folder		
⚙️ .gitignore	9/27/2024 2:33 PM	Git Ignore Source ...	1 KB	
📄 ! docker-compose	9/27/2024 2:33 PM	Yaml Source File	2 KB	
📄 ! docker-compose-v3	9/27/2024 2:33 PM	Yaml Source File	3 KB	
📄 hadoop.env	9/27/2024 2:33 PM	ENV File	3 KB	
📄 Makefile	9/27/2024 2:33 PM	File	2 KB	
📄 README	9/27/2024 2:33 PM	Markdown Source ...	3 KB	

Exécution de la Commande

Après avoir configuré votre environnement Hadoop, vous devez exécuter la commande suivante

```
C:\Users\nourh\docker-hadoop>docker-compose up -d|
```

Cette commande est utilisée pour démarrer et exécuter des conteneurs Docker

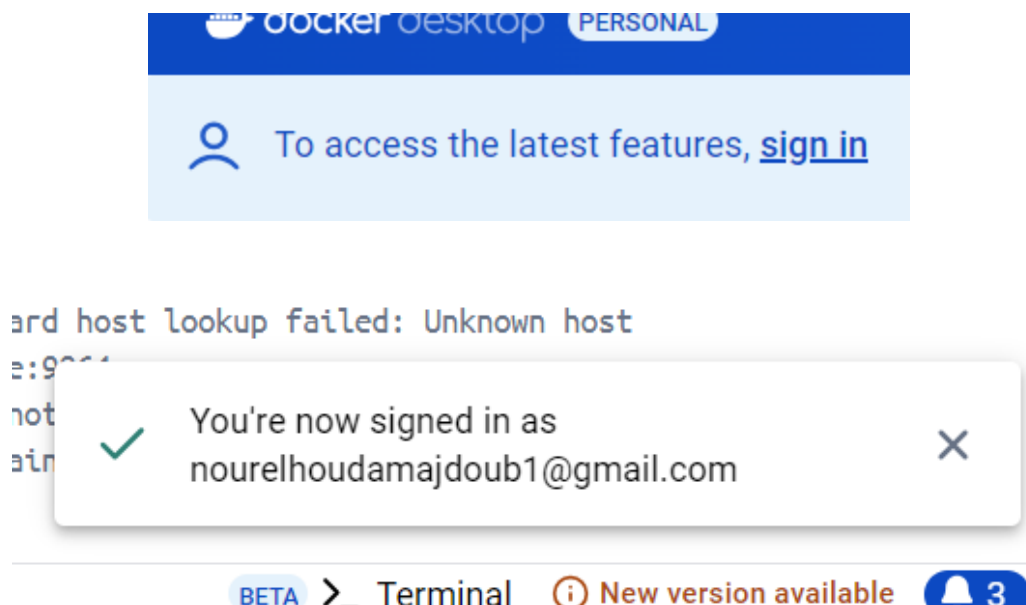
définis dans un fichier

`docker-compose.yml`

Erreur Courante : Problèmes de Configuration de Docker Desktop

Une erreur fréquente que vous pourriez rencontrer est liée à des paramètres incorrects dans l'application Docker Desktop. Cette erreur peut se manifester par des messages indiquant que Docker ne peut pas démarrer correctement ou qu'il y a des problèmes de connexion aux conteneurs

J'ai résolu cette erreur par le sign in dans Docker desktop



Si cette étape est correctement résolue, la sortie dans le terminal devrait ressembler à ceci

```
version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 5/0
✓Container nodemanager      Running      0.0s
✓Container resourcemanager  Running      0.0s
✓Container historyserver    Running      0.0s
✓Container namenode         Running      0.0s
✓Container datanode         Started      0.0s
C:\Users\nourh\docker-hadoop>
```

Cette sortie indique que tous les conteneurs du cluster Hadoop fonctionnent actuellement. Chaque conteneur qui s'affiche dans la liste correspond à un service spécifique dans votre cluster Hadoop, tel que le nœud maître, le nœud de données, ou d'autres services associés. La présence de ces conteneurs signifie

qu'ils sont opérationnels et prêts à traiter des tâches

Partie 2 : Commandes HDFS

Exécution de la Commande

Pour interagir avec le système de fichiers HDFS, j'exécute la commande suivante :

```
docker exec -it namenode bash
```

Après avoir exécuté cette commande, je remarque que mon environnement de travail a changé. Je suis maintenant dans un environnement de conteneur et non plus sur le système hôte.

L'invite de commande affichée après avoir exécuté la commande devrait ressembler à ceci :

```
root@8887807d4366:~#
```

root :

- Cela indique que je suis maintenant connecté en tant qu'utilisateur root. J'ai donc tous les privilèges nécessaires pour exécuter des commandes et effectuer des modifications dans le système de fichiers du conteneur.

@8887807d4366 :

- Ceci est l'identifiant du conteneur actuel dans lequel je me trouve. Cet ID unique permet de savoir précisément dans quel conteneur je travaille, ce qui est utile pour éviter toute confusion entre différents conteneurs.

:/# :

- Cette partie indique le répertoire courant dans lequel je me trouve à l'intérieur du conteneur. Le symbole `#` montre que je suis en tant qu'utilisateur root. L'absence d'un chemin spécifique (par exemple, `/home`) après `:` signifie que je suis dans le répertoire racine du conteneur.

:/# indicates our current directory

```
C:\Users\nourh\docker-hadoop>docker exec -it namenode bash
root@8887807d4366:/# |
```

HDFS

Dans cette section, nous allons nous concentrer sur la manipulation de fichiers à la fois dans le conteneur et sur le système hôte.

Pour commencer, j'ai créé un fichier nommé

`bonjour.txt` à l'intérieur du conteneur en exécutant la commande suivante :

`echo "hi hadoop" > bonjour.txt`

```
root@8958f6a5fbc5:/# ls
KEYS  boot  data  dontgetit  etc  hadoop-data  hi.txt  lib  media  opt  root  run.sh  srv  tmp  var
bin  bye.txt  dev  entrypoint.sh  hadoop  hello.txt  home  lib64  mnt  proc  run  sbin  sys  usr
root@8958f6a5fbc5:/# echo "hi hadoop">bonjour.txt
root@8958f6a5fbc5:/# ls
KEYS  boot  dev  etc  hello.txt  lib  mnt  root  sbin  tmp
bin  bye.txt  dontgetit  hadoop  hi.txt  lib64  opt  run  srv  usr
bonjour.txt  data  entrypoint.sh  hadoop-data  home  media  proc  run.sh  sys  var
```

Où se Trouve ce Fichier ?

Le fichier `bonjour.txt` que j'ai créé se trouve dans le système de fichiers du conteneur. Il **n'est pas** dans HDFS ni sur ma machine locale actuelle. Cela signifie que le fichier est accessible uniquement à l'intérieur du conteneur où il a été créé.

```
root@8958f6a5fbc5:/# hdfs dfs -ls /
Found 1 items
drwxr-xr-x  - root supergroup          0 2024-10-11 09:55 /rmstate
```

Dans l'exemple précédent, nous avons manipulé (en l'occurrence, listé) les fichiers dans HDFS. Maintenant, nous allons essayer de déplacer le fichier `bonjour.txt` que nous avons créé à l'intérieur du conteneur vers HDFS.

```

root@72880e695b03:/# hdfs dfs -put bonjour.txt /
root@72880e695b03:/# hdfs dfs -ls /
Found 3 items
-rw-r--r--    3 root supergroup          0 2024-10-11 12:55 /bonjour.txt
drwxr-xr-x    - root supergroup          0 2024-10-11 12:21 /rmstate
-rw-r--r--    3 root supergroup          0 2024-10-11 12:54 /test.txt
root@72880e695b03:/#

```

Partie3

```

root@8958f6a5fbc5:/# hdfs dfs -mkdir /tps
root@8958f6a5fbc5:/# hdfs dfs -mkdir /tps/data
root@8958f6a5fbc5:/# hdfs dfs -ls
ls: `.`: No such file or directory
root@8958f6a5fbc5:/# hdfs dfs -ls /
Found 3 items
drwxr-xr-x    - root supergroup          0 2024-10-11 10:27 /hdfolder
drwxr-xr-x    - root supergroup          0 2024-10-11 09:55 /rmstate
drwxr-xr-x    - root supergroup          0 2024-10-11 10:50 /tps
root@8958f6a5fbc5:/# |

```

Partie3

Écrivez quelque chose, appuyez sur la touche espace pour valider la commande...

```

C:\Users\nourh\docker-hadoop>docker cp "C:\Users\nourh\Downloads\archive (2)
\p.txt" 38482a65a1aa:/
Successfully copied 211MB to 38482a65a1aa:/
C:\Users\nourh\docker-hadoop>|

```

J'ai renommé purchases to p.txt

```

root@38482a65a1aa:/# hdfs dfs -put p.txt /tps/data|

```

hdfs dfs -ls -R

```
drwxr-xr-x - root supergroup 0 2024-10-11 12:57 /tps
tanode exited with error 0 2024-10-15 12:24 /tps/data
storyserver | -rw-r--r-- 3 root supergroup 0 2024-10-15 12:24 /tps/data/p.txt
```