

# Install MongoDB Community Edition on Ubuntu 20

## Version MongoDB

MongoDB 4.0 Community Edition.

## Support de plate-forme

MongoDB fournit uniquement des packages pour les versions Ubuntu suivantes de LTS (support à long terme) 64 bits 20.04 LTS

### 1. Import the public key used by the package management system

Installer un serveur MongoDB

```
wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo  
apt-key add -
```

- Install gnupg and its required libraries using the following command:

```
sudo apt-get install gnupg
```

- Once installed, retry importing the key:

```
wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo  
apt-key add -
```

### 2. Create a /etc/apt/sources.list.d/mongodb-org-4.2.list file for MongoDB

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu  
focal/mongodb-org/4.4 multiverse" | sudo tee  
/etc/apt/sources.list.d/mongodb-org-4.4.list
```

### 3. Reload local package database

```
sudo apt-get update
```

### 4. Install the MongoDB packages.

```
sudo apt-get install -y mongodb-org  
echo "mongodb-org hold" | sudo dpkg --set-selections  
echo "mongodb-org-server hold" | sudo dpkg --set-selections  
echo "mongodb-org-shell hold" | sudo dpkg --set-selections  
echo "mongodb-org-mongos hold" | sudo dpkg --set-selections  
echo "mongodb-org-tools hold" | sudo dpkg --set-selections
```

mongod

```

nduser@prf-VirtualBox:~$ mongod
{"t":{"$date":"2022-08-31T18:32:01.460+02:00"},"s":"I", "c":"CONTROL", "id":
{"t":{"$date":"2022-08-31T18:32:01.462+02:00"},"s":"I", "c":"NETWORK", "id
enClient, and tcpFastOpenQueueSize."}
{"t":{"$date":"2022-08-31T18:32:01.462+02:00"},"s":"I", "c":"STORAGE", "id
4-bit", "host":"prf-VirtualBox"}}
{"t":{"$date":"2022-08-31T18:32:01.473+02:00"},"s":"I", "c":"CONTROL", "id
874c61d138fd75d6a6f", "opensslVersion":"OpenSSL 1.1.1f 31 Mar 2020", "modules
{"t":{"$date":"2022-08-31T18:32:01.473+02:00"},"s":"I", "c":"CONTROL", "id
{"t":{"$date":"2022-08-31T18:32:01.473+02:00"},"s":"I", "c":"CONTROL", "id
{"t":{"$date":"2022-08-31T18:32:01.474+02:00"},"s":"E", "c":"STORAGE", "id
tory /data/db not found. Create the missing directory or specify another pat
{"t":{"$date":"2022-08-31T18:32:01.474+02:00"},"s":"I", "c":"REPL", "id
{"t":{"$date":"2022-08-31T18:32:01.474+02:00"},"s":"I", "c":"COMMAND", "id
{"t":{"$date":"2022-08-31T18:32:01.474+02:00"},"s":"I", "c":"SHARDING", "id

```

Nous devons créer le répertoire `/data/db/` c'est le répertoire de stockage de la base de données.

```
hduser@prf-VirtualBox:~$ sudo mkdir /data
[sudo] Mot de passe de hduser :
```

Cd /data

```
Sudo chown -R /data
```

```
mkdir /db
```

cd

```
sudo chown -R $USER /data/db
```

Pour lancer le serveur mongodb vous tapez :

mongod

Sur un autre terminal vous pouvez lancer mongo shell en utilisant la commande suivante :

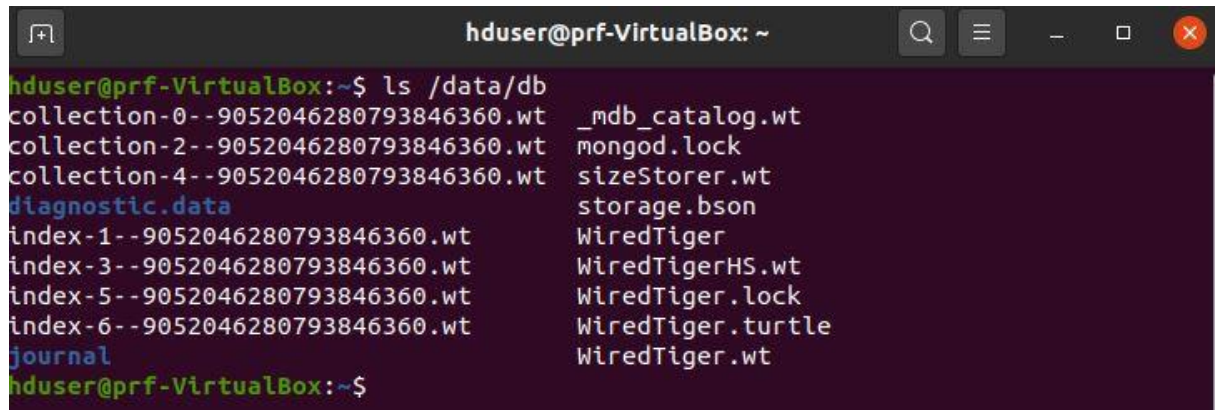
mongo

```
hduser@prf-VirtualBox:~$ mongo
MongoDB shell version v4.4.16
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=
e=mongodb
Implicit session: session { "id" : UUID("03a8cb8c-7329-4175-9d8b-5ceab82e9252")
e}
MongoDB server version: 4.4.16
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
https://docs.mongodb.com/
Questions? Try the MongoDB Developer Community Forums
https://community.mongodb.com
```

```
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
--
> |
```

Pour consulter le repertoire data/db vous pouvez taper la commande suivante :

```
ls /data/db
```



A terminal window titled 'hduser@prf-VirtualBox: ~' showing the output of the command 'ls /data/db'. The output lists various files and directories in the MongoDB data directory, including collection files, diagnostic data, index files, journal, and WiredTiger files.

```
hduser@prf-VirtualBox:~$ ls /data/db
collection-0--9052046280793846360.wt  _mdb_catalog.wt
collection-2--9052046280793846360.wt  mongod.lock
collection-4--9052046280793846360.wt  sizeStorer.wt
diagnostic.data                        storage.bson
index-1--9052046280793846360.wt       WiredTiger
index-3--9052046280793846360.wt       WiredTigerHS.wt
index-5--9052046280793846360.wt       WiredTiger.lock
index-6--9052046280793846360.wt       WiredTiger.turtle
journal                               WiredTiger.wt
hduser@prf-VirtualBox:~$
```

Un ensemble de répertoires et de fichiers ont été créé pour nous permettre de faire tourner la base de données mongodb.

Nous allons pouvoir nous connecter à la base de données mongo en utilisant le port 27017 qui est le port d'écoute de notre serveur et à l'aide de la commande `mongo` on peut lancer mongo shell.

Pour consulter les différentes bases de données disponibles vous utilisez la commande suivante :

```
Show dbs ;
```



A terminal window showing the output of the command 'show dbs;'. The output lists three databases: admin, config, and local, each with a size of 0.000GB.

```
> show dbs;
admin    0.000GB
config  0.000GB
local   0.000GB
```

Les trois bases de données affichées sont les bases de données créées par défaut sur mongodb.

Pour créer une base de données vous tapez :

```
use nom de la base de données ;
```



A terminal window showing the output of the command 'use db1;'. The output is 'switched to db db1'.

```
> use db1;
switched to db db1
```

Si on utilise `show dbs ;` la base de données db1 ne figure pas parce qu'il ne contient pas des données. Pour insérer des données vous utilisez :

```
db.nomCollection.insert({clé :valeurs});
```

Après avoir créer la collection avec des données la base de données figurera dans le résultat de la commande `show dbs ;`

```
> show dbs;
admin    0.000GB
config  0.000GB
local    0.000GB
> db.books.insert({"nom":"mongo book"});
WriteResult({ "nInserted" : 1 })
> show dbs;
admin    0.000GB
config  0.000GB
db1      0.000GB
local    0.000GB
```

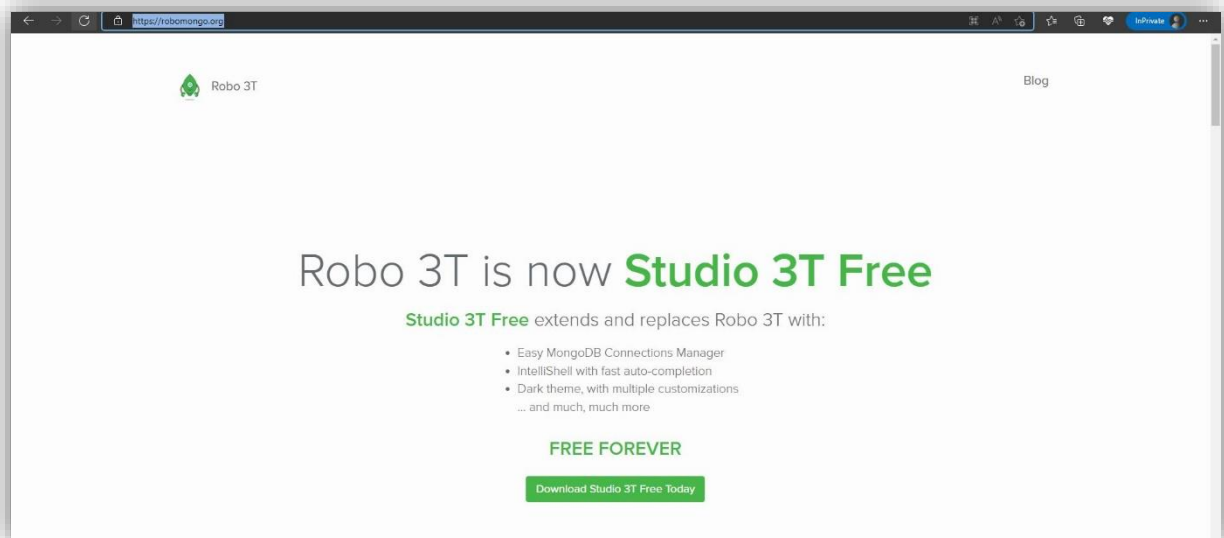
Pour afficher les collections, vous pouvez utiliser `show collections ;`

Pour afficher tous les documents dans une collection : `db.nomCollection.find()`

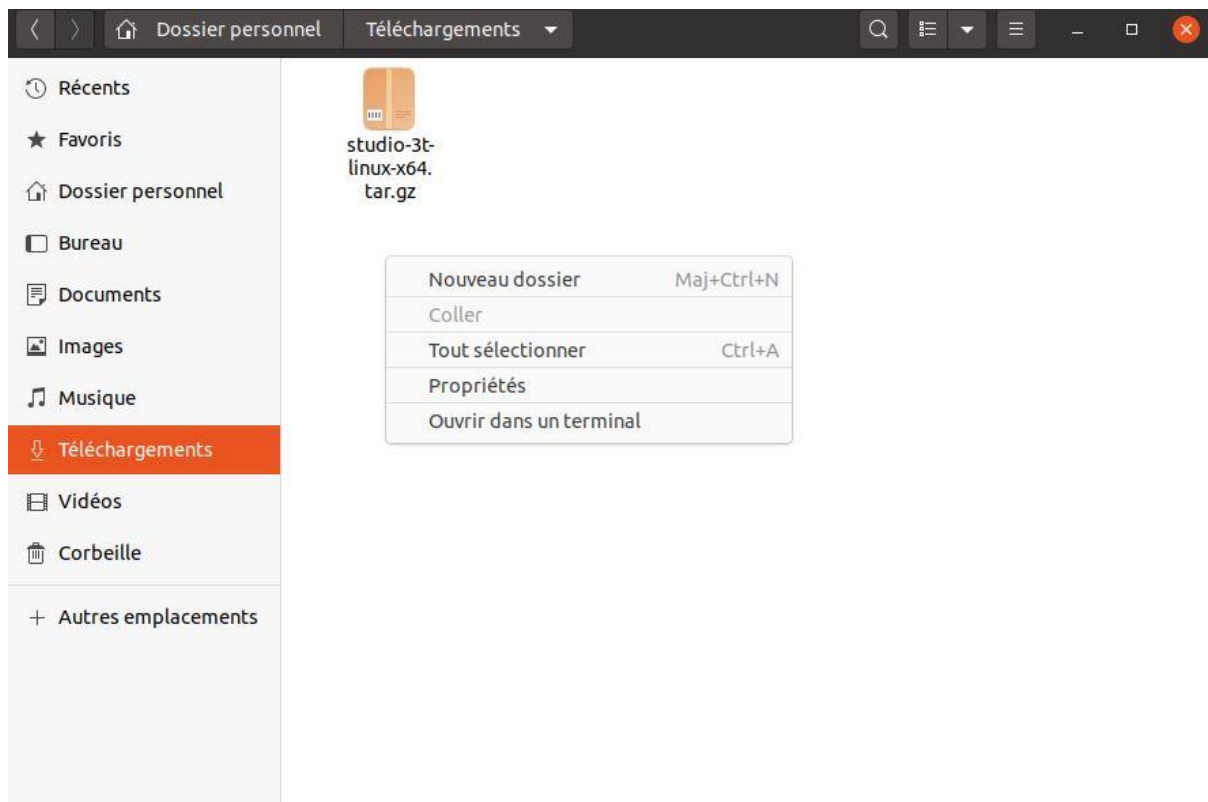
```
> use db1
switched to db db1
> show collections;
books
> db.books.find()
{ "_id" : ObjectId("630fe158ceba88af8c8fae57"), "nom" : "mongo book" }
```

Dans ce TP, nous allons utiliser « Robot 3T » : un interface utilisateur qui va nous faciliter la tâche d'interagir avec mongodb.

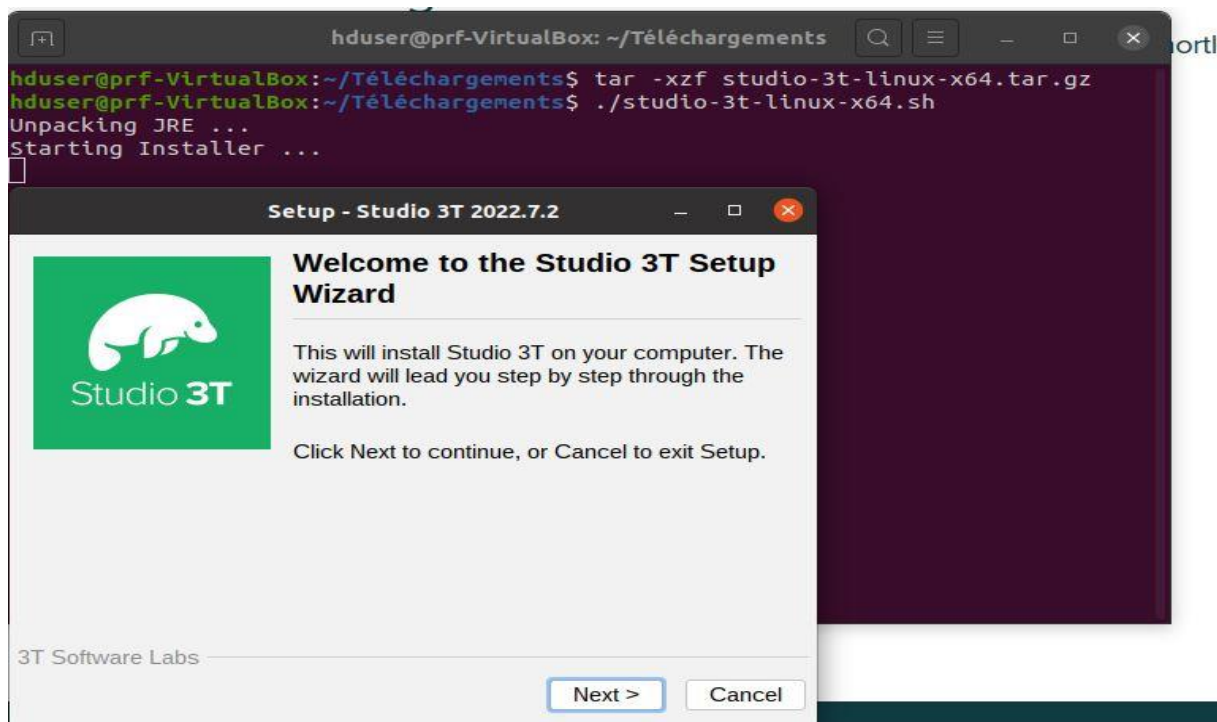
Télécharger Robot 3T en utilisant ce lien : [Robo 3T | Free, open-source MongoDB GUI \(formerly Robomongo\)](https://robomongo.org)



Vous cliquez (click-droit) sur téléchargement, après vous choisissez terminal :



Vous tapez les commandes suivantes pour dézipper le fichier téléchargé et le lancer.





## Set Up Password Encryption

For your convenience, you can store connection passwords in Studio 3T's Connection Manager. Studio 3T will encrypt all passwords before storing them.

Please choose how you would like Studio 3T to encrypt your passwords:

☐ Use Studio 3T's default password encryption


☒ Use cryptographic key store

The use of a cryptographic key store adds extra security. The key store will be protected by a master password that you will be prompted to enter each time Studio 3T launches.

### IMPORTANT:

When switching to the Cryptographic Key Store, Studio 3T imports all connections into the store. The files that previously stored the connections are then deleted to ensure that they are no longer accessible. The connections will only be visible in Studio 3T 2020.5 or later. Turn off the cryptographic key store before moving to a version earlier than that.

Please enter your new master password. You will be asked to enter it each time you launch Studio 3T.

Enter Master Password:  

Confirm Master Password:  

Cancel

Back

Next

Il nous reste qu'à créer un compte gratuit sur studio3T et interagir avec les données.

