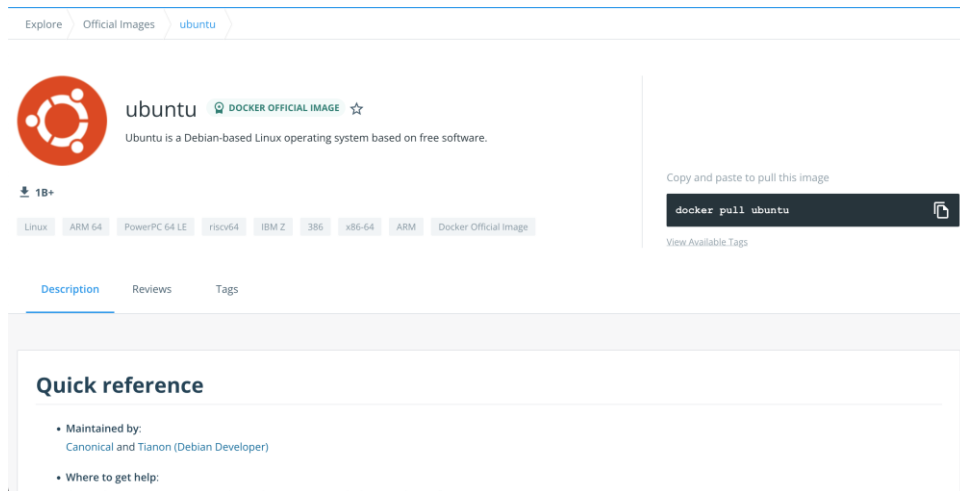


Docker

Docker est une plateforme permettant de lancer des applications dans des conteneurs logiciels. L'utilisateur peut emballer une application et ses dépendances dans un conteneur isolé qui pourra être exécuté sur n'importe quel serveur.

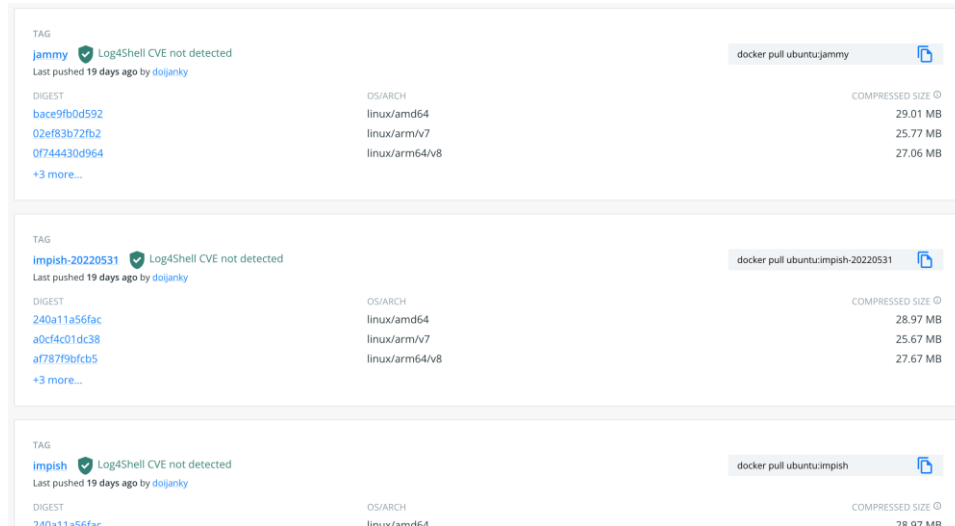
La conteneurisation a l'avantage d'être bien plus légère que la virtualisation. Ce qui rend le conteneur très flexible et portable.

Nous allons créer ensemble un conteneur Ubuntu. Pour cela nous nous rendons d'abord sur Docker Hub afin de consulter l'image à partir de laquelle nous allons créer le conteneur.



The screenshot shows the Docker Hub page for the 'ubuntu' image. At the top, there are tabs for 'Explore', 'Official Images', and 'ubuntu'. The 'ubuntu' tab is selected. Below the tabs, the Ubuntu logo is displayed, followed by the text 'ubuntu' and 'DOCKER OFFICIAL IMAGE'. A description states 'Ubuntu is a Debian-based Linux operating system based on free software.' Below this, there are tabs for 'Description', 'Reviews', and 'Tags'. The 'Tags' tab is selected. On the right side, there is a button to 'Copy and paste to pull this image' with the command 'docker pull ubuntu' and a 'View Available Tags' link. Below the 'Tags' tab, there is a 'Quick reference' section with links to 'Maintained by' (Canonical and Tianon (Debian Developer)) and 'Where to get help' (the Docker Community Forums, the Docker Community Slack, or Stack Overflow).

Nous nous rendons dans les tags afin de sélectionner la version qui nous convient



The screenshot shows the 'tags' page for the 'ubuntu' image. It lists three tags: 'jammy', 'impish-20220531', and 'impish'. Each tag entry includes a 'TAG' section with a checkmark indicating 'Log4Shell CVE not detected', a 'Last pushed' date, and a 'Docker pull' button. Below the 'TAG' section, there is a table with columns for 'DIGEST', 'OS/ARCH', and 'COMPRESSED SIZE'. The 'jammy' tag has three entries for different architectures: 'linux/amd64' (29.01 MB), 'linux/arm/v7' (25.77 MB), and 'linux/arm64/v8' (27.06 MB). The 'impish-20220531' tag has three entries for different architectures: 'linux/amd64' (28.97 MB), 'linux/arm/v7' (25.67 MB), and 'linux/arm64/v8' (27.67 MB). The 'impish' tag has one entry for 'linux/amd64' (28.97 MB).

Nous utilisons la commande **Docker pull** afin de télécharger la dernière version (Jammy Jellyfish) de l'image d'Ubuntu. Il est déconseillé d'utiliser le tag :latest en production car cela peut créer des problèmes de compatibilités lors des mises à jours.

```
➔ ~ docker pull ubuntu:jammy
jammy: Pulling from library/ubuntu
405f018f9d1d: Pull complete
Digest: sha256:b6b83d3c331794420340093eb706a6f152d9c1fa51b262d9bf34594887c2c7ac
Status: Downloaded newer image for ubuntu:jammy
docker.io/library/ubuntu:jammy
➔ ~ |
```

L'image a bien été téléchargée. Nous pouvons l'afficher avec **docker image ls**

```
+ ~ docker image ls
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
bitnami/wordpress	5.9.3-debian-11-r3	4ba5fe293015	10 days ago	799MB
bitnami/mysql	8.0.29-debian-11-r3	d335c66b87ce	10 days ago	431MB
mongo	latest	1d3f6d5230f6	2 weeks ago	696MB
ubuntu	jammy	27941809078c	2 weeks ago	77.8MB
bitnami/mariadb	10.6.8-debian-11-r0	7f344817d5ce	2 weeks ago	342MB
arschm98/gsk-book	1.0	743f0e3dc86e	2 weeks ago	253MB
test	latest	13835f169f9b	2 weeks ago	84.3MB
mongo	<none>	b1f99c7425ee	2 weeks ago	696MB
registry.k8s.io/ingress-nginx/controller	<none>	75bdf78d9d67	3 weeks ago	289MB
nginx	latest	0e901e68141f	3 weeks ago	142MB
alpine	latest	e66264b98777	4 weeks ago	5.53MB
mariadb	10.7	f08f6c1759ee	4 weeks ago	414MB
alpine/git	latest	241890ad72b1	4 weeks ago	38.2MB
hubproxy.docker.internal:5000/docker/desktop-kubernetes	kubernetes-v1.24.0-cni-v0.8.5-critools-v1.17.0-cri-dockerd-v0.2.0-1-debian	c3114e97d326	6 weeks ago	348MB
k8s.gcr.io/kube-apiserver	v1.24.0	529072258ccc	6 weeks ago	130MB
k8s.gcr.io/kube-proxy	v1.24.0	77b49675beae	6 weeks ago	110MB
k8s.gcr.io/kube-controller-manager	v1.24.0	88784fb4ac2f	6 weeks ago	119MB
k8s.gcr.io/kube-scheduler	v1.24.0	e3e07dea73e9	6 weeks ago	51MB
ubuntu	latest	d2e4e1f51132	7 weeks ago	77.8MB
k8s.gcr.io/etcd	3.5.3-0	a6be758cef4c	2 months ago	299MB
diouxx/glipi	latest	9e7dc7707656	2 months ago	280MB
rockylinux	latest	c830f8e8f82b	3 months ago	209MB
k8s.gcr.io/pause	3.7	221177c6882a	3 months ago	711KB
nigelpoulton/gsk-book	1.0	076869facc6c	6 months ago	253MB
mysql	5.6	d43b2d5ddc48	6 months ago	303MB
mongo-express	latest	2d2fb2cab8f8	8 months ago	136MB
wordpress	5.8.1-apache	ccd598bc2c91	8 months ago	616MB
registry.k8s.io/ingress-nginx/kube-webhook-certgen	<none>	c41e9fcdaf5a	8 months ago	47.7MB
k8s.gcr.io/coredns/coredns	v1.8.6	a4ca41631cc7	8 months ago	46.8MB
centos	latest	5d06a3dc9764	9 months ago	231MB
docker/desktop-vpnkit-controller	v2.0	8c2c38aa676e	13 months ago	21MB
docker/desktop-storage-provisioner	v2.0	99f89471f470	14 months ago	41.9MB
nigelpoulton/gsk-book	1.1	fa3b534df65e	17 months ago	121MB
nigelpoulton/k8sbook	1.0	591a63f2cae6	3 years ago	73.6MB
wordpress	4.8-apache	7cf3e41b8864	4 years ago	408MB
mongo	3.1	199e537da3a8	6 years ago	303MB

```
+ ~ |
```

Nous allons maintenant créer un conteneur à partir de l'image.

```
➔ ~ docker run -it ubuntu:jammy bin/bash
root@71c7aa34b32a:/# |
```

La commande `docker run` demande à Docker de lancer le conteneur à partir de l'image indiquée. Les paramètres `-it` nous permettent d'avoir une session interactive dans notre terminal. `bin/bash` indique quel langage shell nous souhaitons utiliser.-

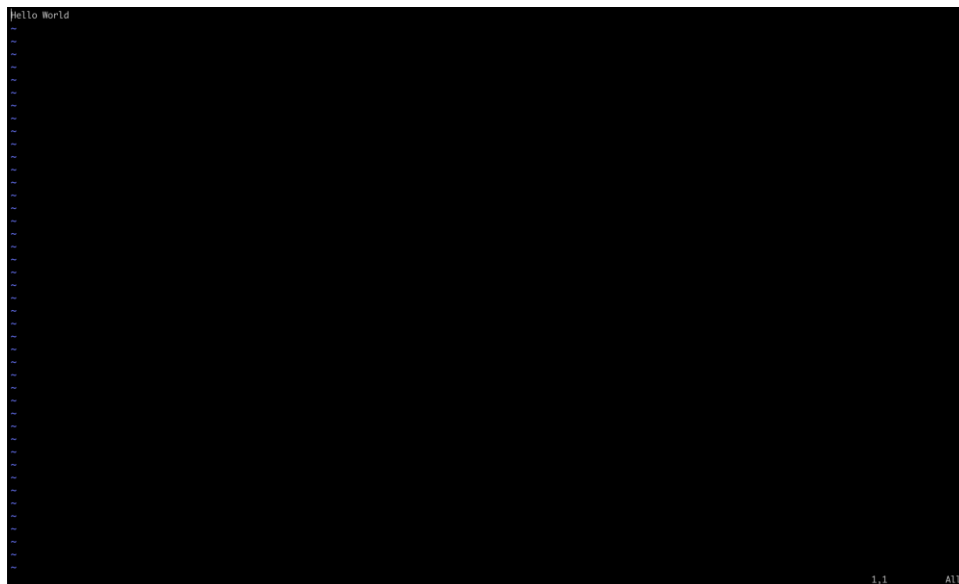
Nous sommes maintenant dans un conteneur Ubuntu. Il s'agit d'une version ultralégère sans interface graphique.

Nous allons mettre à jour nos paquets et installer vim dans ce conteneur.

```
root@71c7aa34b32a:/# apt update && apt upgrade && apt install vim|
```

Nous pouvons vérifier que vim a bien été installé en ouvrant un fichier avec.

```
root@71c7aa34b32a:/# vim /tmp/Hello-World.txt|
```



Le fichier s'ouvre, vim fonctionne correctement.

Nous allons maintenant supprimer le conteneur que nous venons de créer.

Pour voir la liste de tous les conteneurs même éteints nous utilisons **docker ps -a**

```
+ ~ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
71c7aa34b32a	ubuntu:jammy	"bin/bash"	38 minutes ago	Exited (137) 2 minutes ago		goofy_grothendieck
3d557c80774e	dlwuxx/glipi	"/opt/glipi-start.sh"	24 hours ago	Up 24 hours	0.0.0.0:80->80/tcp, 443/tcp	glipi
4ff8c6d76a04	mariaadb:10.7	"docker-entrypoint.s..."	24 hours ago	Up 24 hours	3306/tcp	mariaadb

```
+ ~ |
```

Nous pouvons supprimer le conteneur arrêté en utilisant **docker rm <id>**

```
→ ~ docker rm 71c7aa34b32a
71c7aa34b32a
```

Notre conteneur a bien été supprimé.

```
→ ~ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3d557c80774e	diouxx/glipl	"/opt/glipl-start.sh"	25 hours ago	Up 25 hours	0.0.0.0:80->80/tcp, 443/tcp	glipl
4ff8c6d776a4	mariadb:10.7	"docker-entrypoint.s..."	25 hours ago	Up 25 hours	3306/tcp	mariadb

```
→ ~ |
```