
Configuration d'un nœud Master Kubernetes sur Debian 12 avec containerd

Configurer un nœud master Kubernetes sous Debian 12 en utilisant containerd comme runtime de conteneur.

[DevOps](#) [Systèmes & Réseaux](#) [20 min de lecture](#) [Niveau Intermédiaire](#)

Document généré le 11/07/2026 à 20h42 · nouv.fr/wiki/master-kubernetes-debian12-containerd

Sommaire

12 section(s) · 20 min de lecture

- 1. Mise à jour du système et installation des outils de base**
 - 2. Installation de containerd**
 - 3. Désactivation du swap**
 - 4. Configuration des modules du noyau et paramètres réseau**
 - 5. Ajouter le dépôt Kubernetes et la clé GPG**
 - 6. Installer kubelet, kubeadm et kubectl**
 - 7. Initialiser le cluster Kubernetes**
 - 8. Configurer kubectl pour l'utilisateur courant**
 - 9. Installer le plugin réseau Flannel**
 - 10. Vérifier l'état du nœud Master**
 - 11. Dépannage — Flannel / CNI**
- Récapitulatif — Ordre d'installation**

1. Mise à jour du système et installation des outils de base

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y curl gnupg2 apt-transport-https ca-certificates software-properties-
common
```

📄 Copier

2. Installation de containerd

```
sudo apt install -y containerd
sudo mkdir -p /etc/containerd
sudo containerd config default | sudo tee /etc/containerd/config.toml
sudo sed -i 's/SystemdCgroup = false/SystemdCgroup = true/' /etc/containerd/config.toml
sudo systemctl restart containerd
sudo systemctl enable containerd
sudo systemctl status containerd
```

📄 Copier

L'option `SystemdCgroup = true` est obligatoire pour que containerd s'intègre correctement avec kubelet via systemd.

3. Désactivation du swap

Le swap doit être désactivé — Kubernetes ne fonctionne pas correctement avec le swap actif.

```
sudo swapoff -a
sudo sed -i '/ swap / s/^(.*)$/#1/g' /etc/fstab
```

📄 Copier

*La commande `sed` commente la ligne swap dans `/etc/fstab` pour que la désactivation soit **persistante après redémarrage**.*

4. Configuration des modules du noyau et paramètres réseau

```
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
overlay
br_netfilter
EOF

sudo modprobe overlay
sudo modprobe br_netfilter
```

📄 Copier

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-iptables = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward = 1
EOF

sudo sysctl --system
```

📄 Copier

Paramètre	Rôle
overlay	Système de fichiers en couches pour les conteneurs
br_netfilter	Permet au bridge réseau de passer par iptables
ip_forward	Active le routage IP entre les interfaces

5. Ajouter le dépôt Kubernetes et la clé GPG

```
sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.32/deb/Release.key |
  sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg

echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]
https://pkgs.k8s.io/core:/stable:/v1.32/deb/ /" |
sudo tee /etc/apt/sources.list.d/kubernetes.list
```

📄 Copier

6. Installer kubelet, kubeadm et kubectl

```
sudo apt update
sudo apt install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
```

📄 Copier

apt-mark hold empêche une mise à jour automatique des composants Kubernetes, ce qui pourrait casser le cluster.

7. Initialiser le cluster Kubernetes

```
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
```

📋 Copier

*Le CIDR 10.244.0.0/16 est recommandé pour le plugin réseau **Flannel**.*

À la fin de l'initialisation, kubeadm affiche une commande `kubeadm join` à conserver pour ajouter des nœuds workers.

8. Configurer kubectl pour l'utilisateur courant

```
mkdir -p $HOME/.kube  
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

📋 Copier

9. Installer le plugin réseau Flannel

```
kubectl apply -f  
https://raw.githubusercontent.com/flannel-io/flannel/master/Documentation/kube-flannel.yml
```

📋 Copier

10. Vérifier l'état du nœud Master

```
kubectl get nodes
```

📋 Copier

Résultat attendu :

NAME	STATUS	ROLES	AGE	VERSION
master	Ready	control-plane	5m	v1.32.0

✂ Copier

11. Dépannage — Flannel / CNI

Si les pods Flannel restent en `CrashLoopBackOff` ou que le nœud reste en `NotReady`, c'est souvent lié à un chemin CNI manquant. Créer le lien symbolique suivant :

```
sudo ln -s /opt/cni/bin /usr/lib/cni
```

✂ Copier

Puis redémarrer les pods Flannel :

```
kubectl rollout restart daemonset/kube-flannel-ds -n kube-flannel
kubectl get pods -n kube-flannel
```

✂ Copier

Récapitulatif — Ordre d'installation

Étape	Action
1	Mettre à jour le système et installer les dépendances
2	Installer et configurer containerd (<code>SystemdCgroup = true</code>)
3	Désactiver le swap (temporaire + persistant via <code>fstab</code>)
4	Charger les modules noyau et paramètres réseau
5	Ajouter le dépôt Kubernetes v1.32
6	Installer kubelet, kubeadm, kubectl (avec hold)
7	Initialiser le cluster (<code>kubeadm init</code>)
8	Configurer kubectl pour l'utilisateur
9	Installer Flannel
10	Vérifier <code>kubectl get nodes</code> → STATUS Ready
11	Si Flannel KO : <code>sudo ln -s /opt/cni/bin /usr/lib/cni</code>