



# Module SAE 503

Cahier d'installation technique

Orchestrer la conteneurisation  
d'une application

Elaboré par

**Daniel Aronn**  
**Sérandour Clément**

Promotion

FA3A

Année

Universitaire 2024-2025

<b>Lot 2 : Mise en place</b>	<b>3</b>
1. installation K3s	3
2. Installation Docker	4

## Lot 2 : Mise en place

Sur proxmox nous avons une VM debian sur laquelle nous allons travailler, son ip est 172.18.53.104 sur le serveur 172.19.11.208:8006

Malheureusement suite a un bug non résolue sur la VM Proxmox, nous somme aller sur une VM sous Virtualbox

Spec: OS = ubuntu + K3s

RAM = 2 GO

CPU = 1

Stockage = 20 GO

Sur cette VM nous allons installer deux service, K3s et docker

Nous allons utiliser **K3s**, qui est une solution légère et simple permettant de créer un cluster Kubernetes local sur une machine virtuelle. K3s est parfait pour des environnements de développement et de test.

Pour faire cela on s'aide des informations du site suivant pour l'installation

<https://docs.k3s.io/installation>

### 1. installation K3s

Dans un premier temps, on va installer K3s. Pour cela on utilisera la commande :

**`curl -sfL https://get.k3s.io | sh -`**

Ensuite on peut vérifier que l'installation s'est correctement déroulée avec la commande :

**`sudo k3s kubectl cluster-info`**



## Architecture de l'orchestrateur

L'architecture de l'orchestrateur K3s sera basée sur les mêmes principes que pour un cluster Kubernetes classique, avec les éléments suivants :

- **K3s Cluster** : Comprend un seul nœud maître (et éventuellement des nœuds de travail, mais dans K3s, il est généralement mononucléaire par défaut).
- **Traefik** : Nous utiliserons Traefik comme reverse proxy pour gérer les entrées HTTP et acheminer les requêtes vers les microservices.
- **Microservices** : Chaque point d'accès de l'application sera déployé dans un conteneur distinct (par exemple, /users, /quotes, et /search).

Pour faire cela on s'aidera des informations du site suivant :

<https://kubernetes.io/fr/docs/tasks/tools/install-k3s/>

**sudo usermod -aG docker \$USER && newgrp docker**