**Brief Empowering SUN IT Administrators to Manage Agile and Flexible Infrastructure**

1. **Définition de Kubernetes**

Kubernetes, souvent abrégé en "K8s", est une plateforme open-source d'orchestration de conteneurs. Il a été développé par Google et est maintenant maintenu par la Cloud Native Computing Foundation (CNCF). Kubernetes permet de gérer automatiquement le déploiement, la mise à l'échelle, la gestion des ressources et la gestion de la disponibilité des applications conteneurisées. Il offre un environnement robuste pour le déploiement et la gestion d'applications distribuées et scalables.

1. **Architecture de Kubernetes**

L'architecture de Kubernetes se compose de plusieurs composants principaux, notamment :

* **Le Master Node (nœud maître)** : Le cerveau de la grappe Kubernetes. Il gère l'ensemble du système et prend des décisions sur la répartition des charges, le déploiement des conteneurs, etc.
* **Les Worker Nodes (nœuds de travail)** : Ces nœuds exécutent les conteneurs et gèrent leur exécution. Ils sont responsables de l'exécution des charges de travail et de la communication avec le Master Node.
* **Etcd** : Un magasin de clés-values distribué qui stocke la configuration et l'état de la grappe Kubernetes.
* **Kubelet** : Un agent qui s'exécute sur chaque nœud de travail et communique avec le Master Node. Il s'assure que les conteneurs sont exécutés dans un pod.
* **Kube Proxy** : Un proxy réseau qui gère la communication réseau entre les pods.

1. **Définition d'un Node**

Un "Node" (nœud) dans Kubernetes est une machine physique ou virtuelle sur laquelle les conteneurs s'exécutent. Il fait partie de la grappe Kubernetes et est également appelé "Worker Node" ou "Minion". Chaque nœud a un agent Kubelet qui communique avec le Master Node et s'assure que les conteneurs sont en cours d'exécution conformément à la configuration.

**4. Définition d'un Pod**

Un "Pod" est l'unité de base dans Kubernetes. Il s'agit d'un groupe de un ou plusieurs conteneurs qui partagent le même espace réseau et les mêmes ressources de stockage. Les conteneurs à l'intérieur d'un pod peuvent communiquer les uns avec les autres via localhost, ce qui les rend adaptés à l'exécution d'applications qui nécessitent une forte interaction.

**5. Principales Étapes pour Réaliser un Déploiement**

Pour déployer une application avec Kubernetes, voici les principales étapes à suivre :

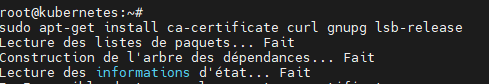
1. **Création d'un fichier YAML** : Définissez les spécifications de votre application, y compris les pods, les services, et les déploiements, dans un fichier YAML.
2. **Création d'une Grappe Kubernetes** : Configurez un cluster Kubernetes en installant un Master Node et plusieurs Worker Nodes.
3. **Création d'un Déploiement** : Utilisez un fichier YAML pour créer un déploiement qui décrit comment votre application doit être déployée, mis à l'échelle, et géré.
4. **Déploiement**  : Appliquez le fichier YAML en utilisant la commande **kubectl apply** pour déployer votre application dans la grappe Kubernetes.
5. **Surveillance et Gestion** : Utilisez des outils comme **kubectl** pour surveiller l'état de votre application et effectuer des opérations de gestion telles que la mise à l'échelle, la mise à jour, et la réparation.
6. **Définition de la Haute Disponibilité**

La "Haute Disponibilité" (High Availability, HA) est un concept qui se réfère à la capacité d'un système à rester opérationnel et accessible en permanence, même en cas de défaillance matérielle ou logicielle. Dans le contexte de Kubernetes, cela signifie garantir que les applications sont toujours disponibles, même si des nœuds individuels échouent. Kubernetes offre des mécanismes de réplication et de gestion de la disponibilité des pods pour garantir la haute disponibilité des applications.

1. **Définition d'un Fichier YAML**

Un fichier YAML (YAML Ain't Markup Language) est un format de fichier de configuration qui est largement utilisé dans l'écosystème Kubernetes pour définir la configuration des ressources, telles que les pods, les services, les déploiements, etc. Les fichiers YAML sont utilisés pour décrire comment les ressources doivent être créées, configurées et gérées dans un cluster Kubernetes. Ils sont lisibles par l'homme et sont également interprétables par des outils de gestion comme **kubectl**.

1. Installation de docker et kubernetes

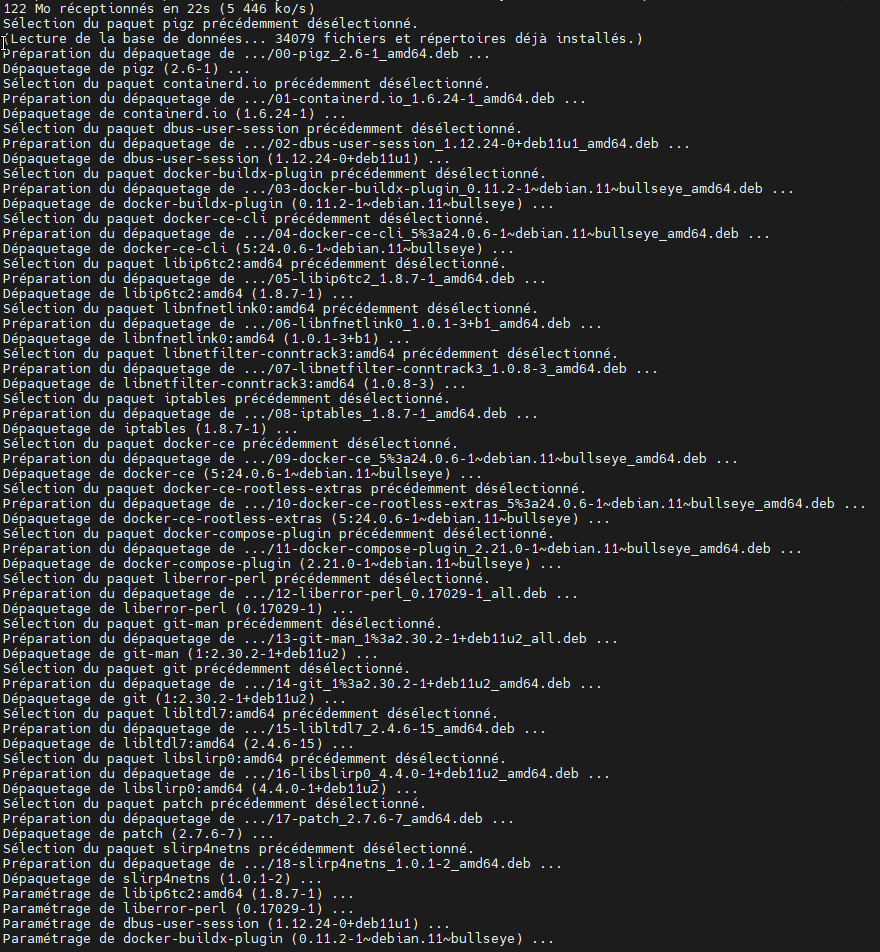


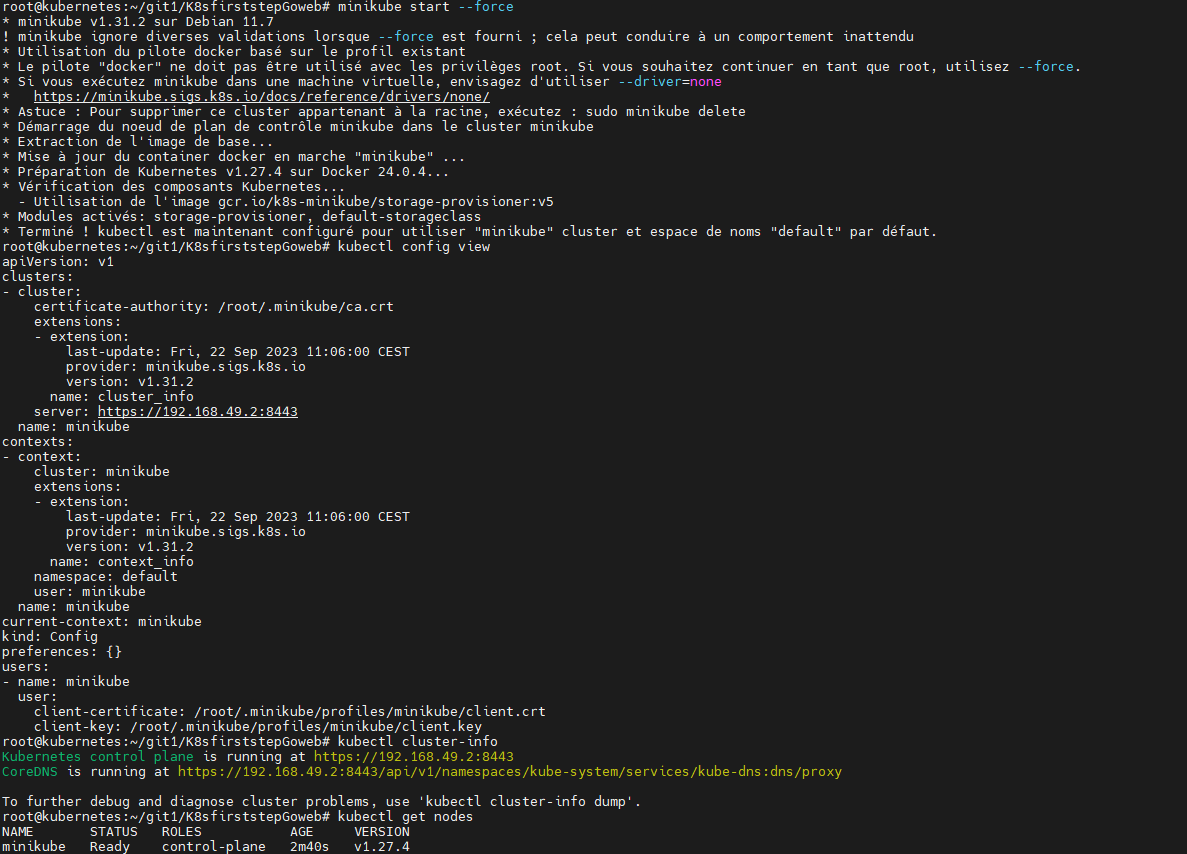
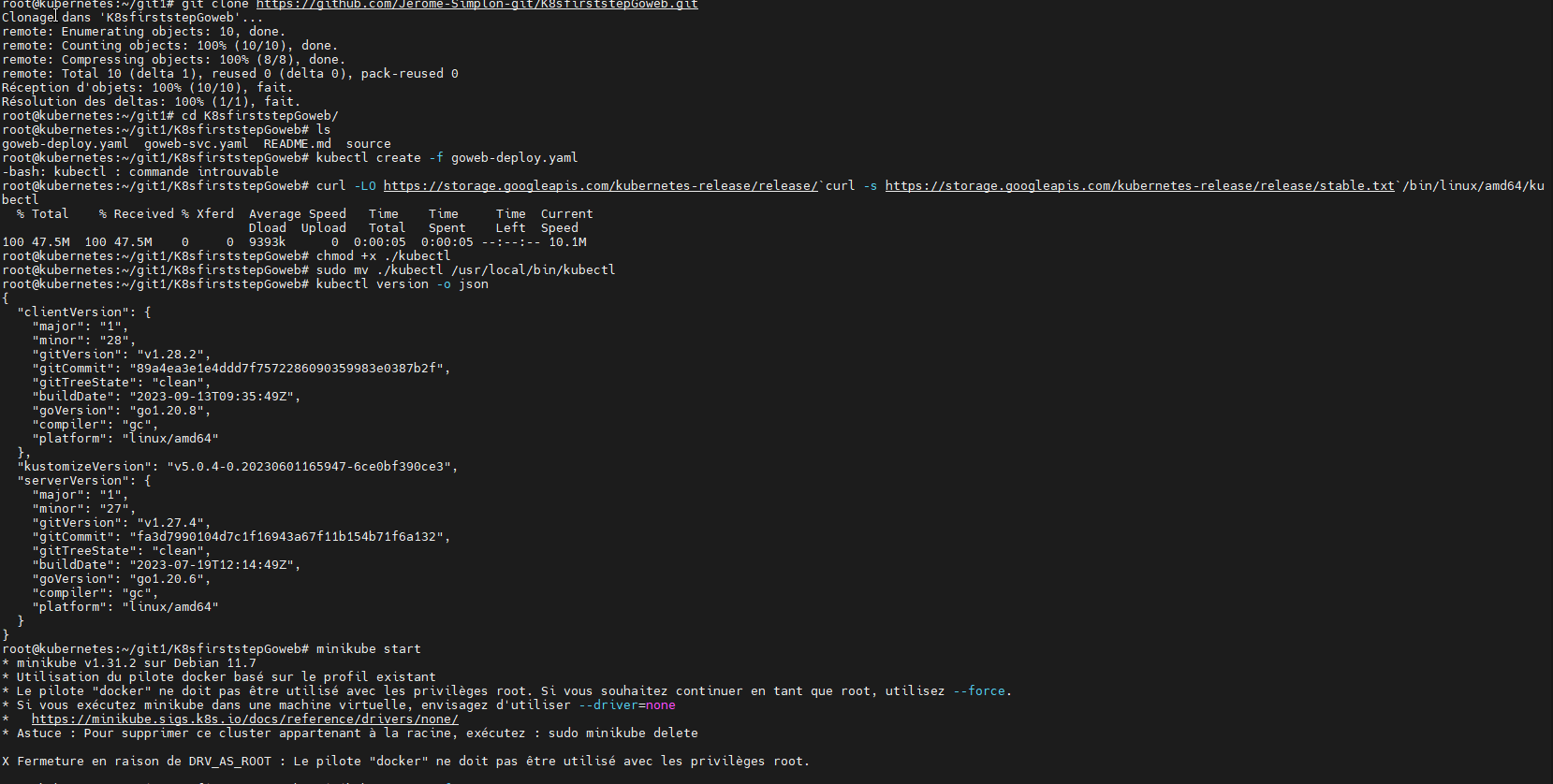
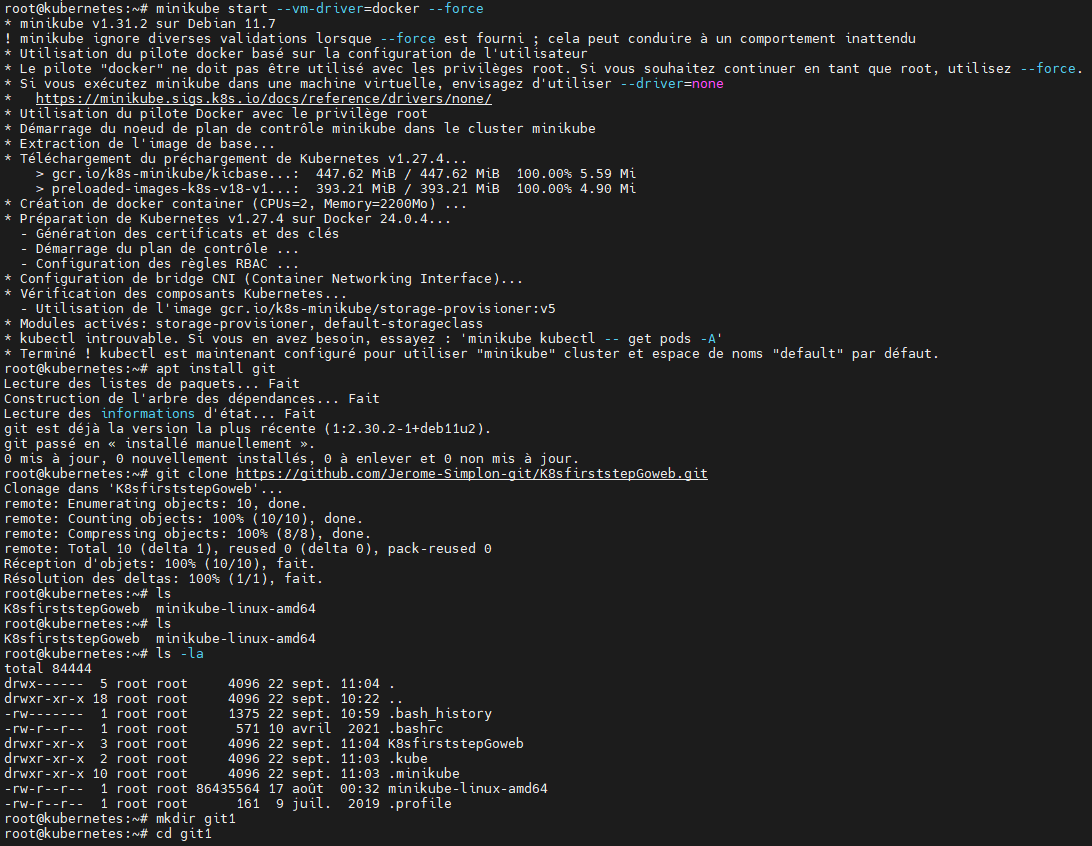
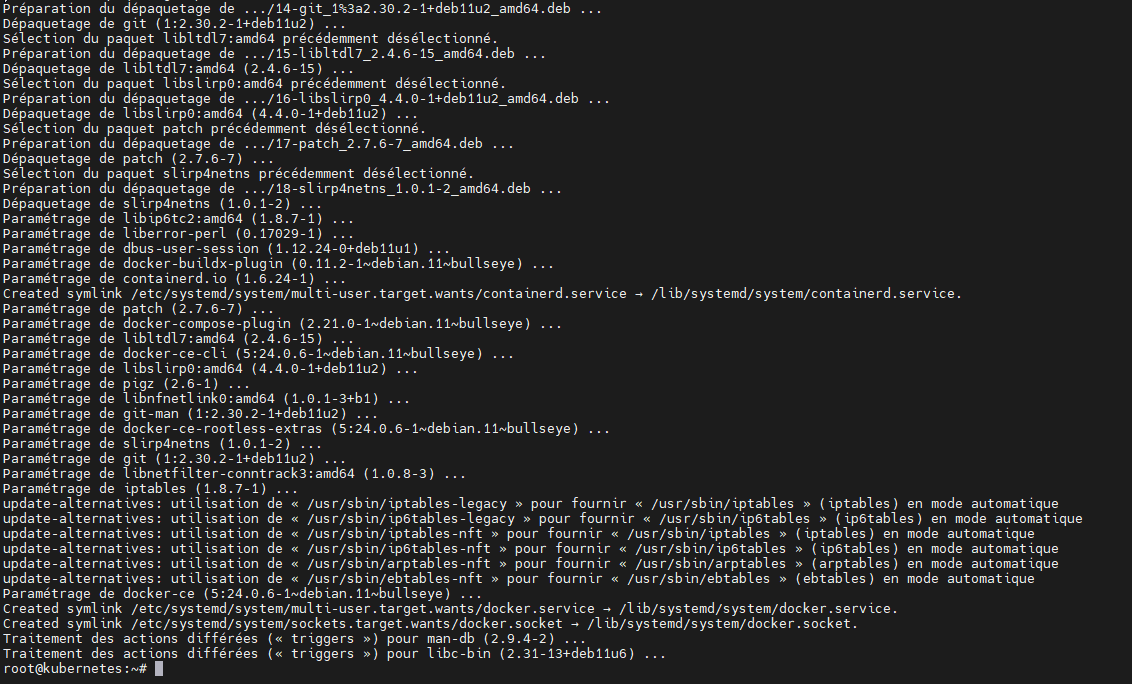
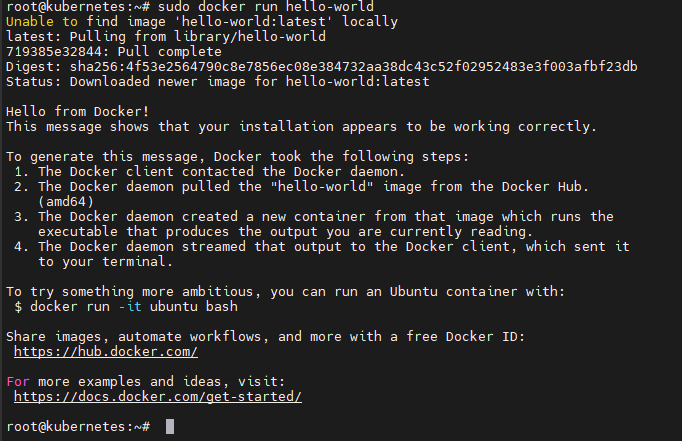


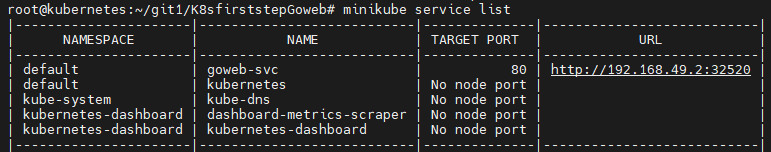
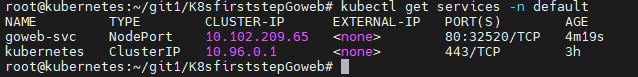
Une image contenant capture d’écran, texte

Description générée automatiquementUne image contenant texte, capture d’écran, Police

Description générée automatiquement







Une image contenant texte, Police, logiciel, nombre

Description générée automatiquement



Une image contenant capture d’écran, texte, ligne, Police

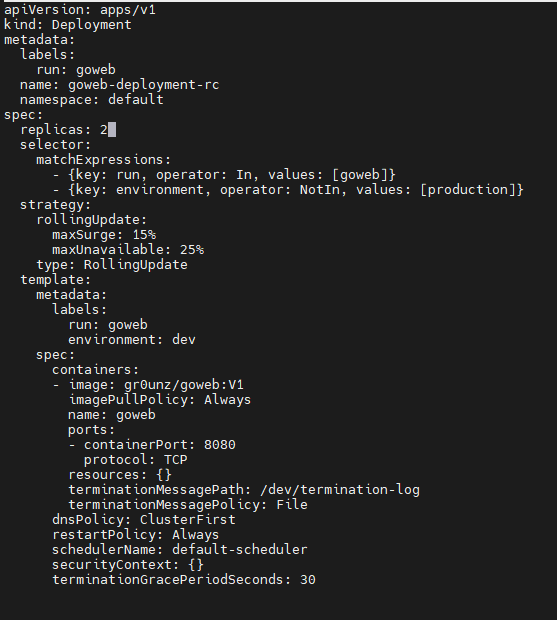
Description générée automatiquement

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquementUne image contenant texte, capture d’écran, Police

Description générée automatiquement



1. Réplica