**Kubernetes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Command** | | |
| **Checking the location of the CoreDNS**  **And where the Kubernetes control plane is running at** | **kubectl cluster-info** | | |
| **Diagnose cluster problems** | **kubectl cluster-info dump** | | |
| **Get information about the cluster** | **Kubectl config get-contexts** | | |
| **test connectivity to the cluster by running** | **kubectl get nodes** | | |
| **Set to another docker context** | **kubectl config set-context docker-desktop** | | |
| **Kubernetes dashboard set up** | **kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml** | | |
| **Port forwarding for visualization** | **Kubectl proxy** | | |
| **Visualization** | [**http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/**](http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/)  **http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/#/login** | | |
| **Next step is to get the token, which you start by opening a code editor** |  | | |
| **Create a yaml directory** | **New-Item -Path "dashboard-adminuser.yaml" -ItemType "file"** | | **apiVersion: v1**  **kind: ServiceAccount**  **metadata:**  **name: admin-user**  **namespace: kubernetes-dashboard** |
|  | **New-Item -Path "dashboard-clusterrole.yaml" -ItemType "file"** | **apiVersion: rbac.authorization.k8s.io/v1**  **kind: ClusterRoleBinding**  **metadata:**  **name: admin-user**  **roleRef:**  **apiGroup: rbac.authorization.k8s.io**  **kind: ClusterRole**  **name: cluster-admin**  **subjects:**  **- kind: ServiceAccount**  **name: admin-user**  **namespace: kubernetes-dashboard**  **I changed the code to**  **apiVersion: rbac.authorization.k8s.io/v1**  **kind: ClusterRoleBinding**  **metadata:**  **name: admin-user-binding  # Make sure this is not empty and unique**  **subjects:**  **- kind: ServiceAccount**  **name: admin-user**  **namespace: kube-system**  **roleRef:**  **kind: ClusterRole**  **name: cluster-admin**  **apiGroup: rbac.authorization.k8s.io** | |
|  | **New-Item -Path "dashboard-secret.yaml" -ItemType "file"** | **apiVersion: v1**  **kind: Secret**  **metadata:**  **name: admin-user**  **namespace: kubernetes-dashboard**  **annotations:**  **kubernetes.io/service-account.name: "admin-user"**  **type: kubernetes.io/service-account-token** | |
| **On the terminal, run** | **kubectl apply**  **-f dashboard-adminuser.yaml** | **Output: serviceaccount/admin-user created** | |
|  | **kubectl apply -f dashboard-clusterrole.yaml** | **Previous code:**  **apiVersion: v1**  **kind: Secret**  **metadata:**  **name: admin-user**  **namespace: kubernetes-dashboard**  **annotations:**  **kubernetes.io/service-account.name: "admin-user"**  **type: kubernetes.io/service-account-tokenkubectl apply --dry-run=client -f dashboard-adminuser.yaml**  **Changed it to:**  **apiVersion: v1**  **kind: Secret**  **metadata:**  **name: admin-user**  **namespace: kubernetes-dashboard**  **annotations:**  **kubernetes.io/service-account.name: "admin-user"**  **type: kubernetes.io/service-account-token** | |
|  |  |  | |
|  |  |  | |
| **list all existing ClusterRoleBindings** | **kubectl get clusterrolebinding** | | |
| **Get k8 token by running** | **kubectl get secret admin-user -n kubernetes-dashboard -o jsonpath="{.data.token}"** | | |
|  |  | | |
| **If the token is not working and outputs a 401 error,** | 1. **Service Account and Namespace:**  * **Confirm that the ServiceAccount admin-user is in the kubernetes-dashboard namespace.** * **Ensure that the ClusterRoleBinding correctly binds the admin-user ServiceAccount to the cluster-admin role.**   **You can verify these using the following commands:**  **kubectl get serviceaccount admin-user -n kubernetes-dashboard**  **kubectl get clusterrolebinding admin-user-binding**   1. **Check Secret and Token:**  * **Verify that the token stored in the Secret is valid and matches the expected token format.**   **You can check the Secret directly with:**  **kubectl describe secret admin-user -n kubernetes-dashboard** | | |
| **1. Check Dashboard Permissions**  **Ensure that the ServiceAccount you’re using has the necessary permissions to view resources in the cluster. Since you've assigned the cluster-admin role, this should grant full access. Confirm the ClusterRoleBinding is correctly set up** | **kubectl get clusterrolebinding admin-user-binding -o yaml** | | |
| **Installing helm** | **curl https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash** | | |
| **Installing apache airflow** | **First add it to the helm repository: helm repo add apache-airflow** [**https://airflow.apache.org**](https://airflow.apache.org)  **Then:** **helm install airflow apache-airflow/airflow --namespace airflow --create-namespace** | | |
| **Cleanining up the resource to redownload it again. In this case, airflow** | **helm uninstall airflow -n airflow**  **kubectl delete namespace airflow** | | |
| **Check connection error for airflow on why localhost:8080 is not working** | **kubectl port-forward svc/airflow-webserver 8080:8080 –namespace airflow**  **kubectl get pods -n airflow** | | |
| **Shows apache value configurations** | **helm show values apache-airflow/airflow > values.yaml** | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |
|  |  | | |

**helm install airflow apache-airflow/airflow --namespace airflow --create-namespace –debug**