

**Lab Manual- Create a Windows Server container on an Azure Kubernetes Service (AKS) cluster using the Azure portal**

**Prepared for**:

**Date:** 18th Dec 2023

**Prepared by:**

Document Name: Lab Manual **Document Number** AZLabn916

**Contributor:**

Contents

[1. Objective 3](#_Toc155432767)

[2. Use the default bridge network 3](#_Toc155432768)

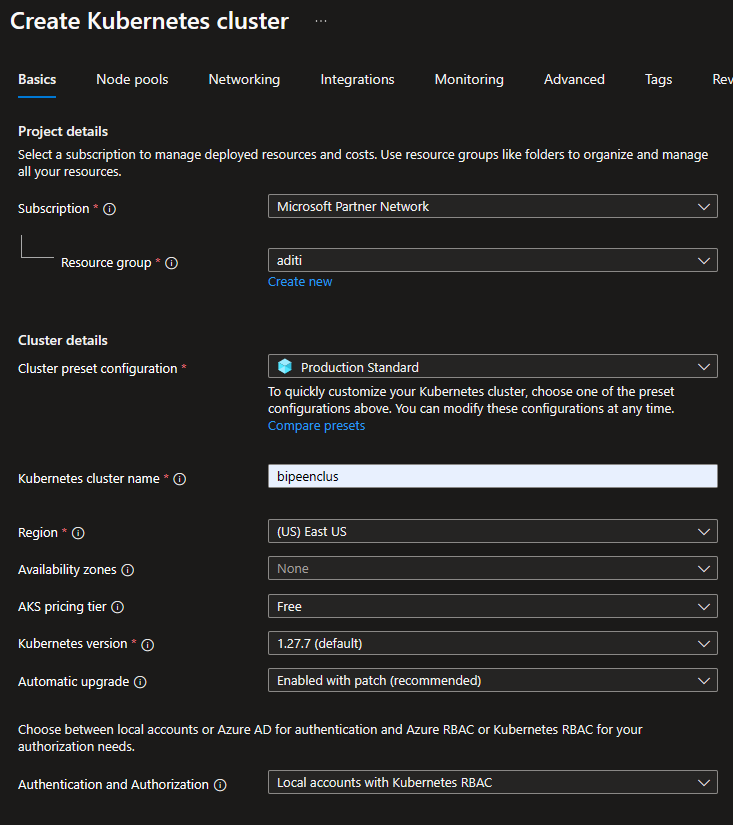
[3. Use the User-defined bridge network 6](#_Toc155432769)

# Objective

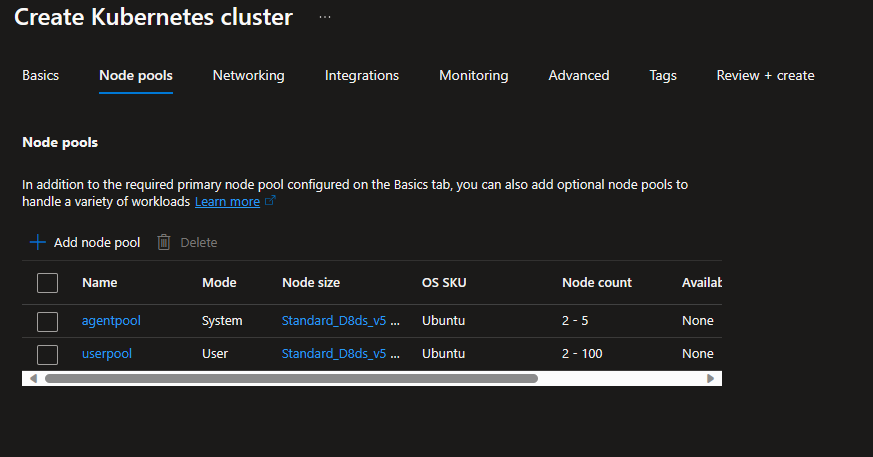
Azure Kubernetes Service (AKS) is a managed Kubernetes service that lets you quickly deploy and manage clusters. In this article, you deploy an AKS cluster that runs Windows Server containers using the Azure portal. You also deploy an ASP.NET sample application in a Windows Server container to the cluster.

# Create an AKS cluster

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. On the Azure portal home page, select **Create a resource**.
3. In the **Categories** section, select **Containers** > **Azure Kubernetes Service (AKS)**.
4. On the **Basics** tab, configure the following options:
   * Under **Project details**:
     + Select an Azure **Subscription**.
     + Select an Azure **Resource group**.
   * Under **Cluster details**:
     + Set the **Cluster preset configuration** to Production Standard.
5. Enter a **Kubernetes cluster name**, such as myAKSCluster.
6. Select a **Region** for the AKS cluster.
7. Leave the **Availability zones** setting set to None.
8. Leave the **AKS pricing tier** setting set to Standard.
9. Leave the default value selected for **Kubernetes version**.
10. Leave the **Automatic upgrade** setting set to the recommended value, which is Enabled with patch.
11. Leave the **Authentication and authorization** setting set to Local accounts with Kubernetes RBAC.

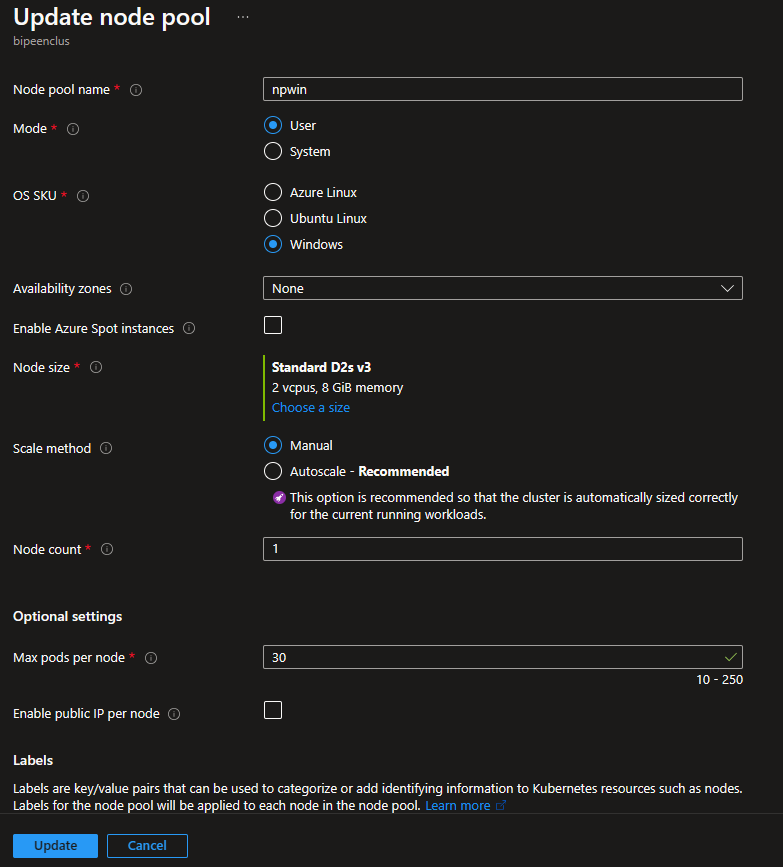


1. Select **Next**. On the **Node pools** tab,



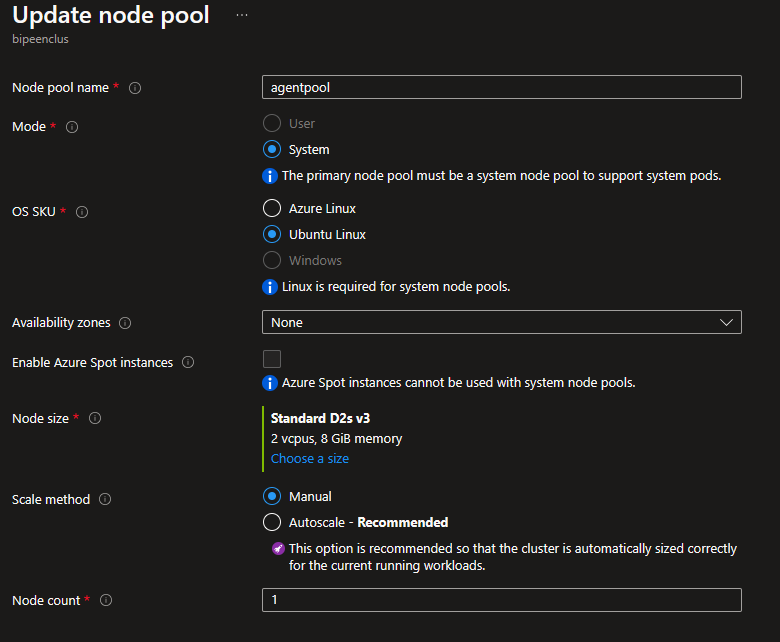
1. Add a new node pool:

* Select **Add node pool**.
* Enter a **Node pool name**, such as npwin. For a Windows node pool, the name must be six characters or fewer.
* For **Mode**, select **User**.
* For **OS SKU**, select **Windows**.
* Leave the **Availability zones** setting set to None.
* Leave the **Enable Azure Spot instances** checkbox unchecked.
* For **Node size**, select **Choose a size**. On the **Select a VM size** page, select D2s\_v3, then choose the **Select** button.
* Leave the **Scale method** setting set to Manual
* Set the Node Count to 1

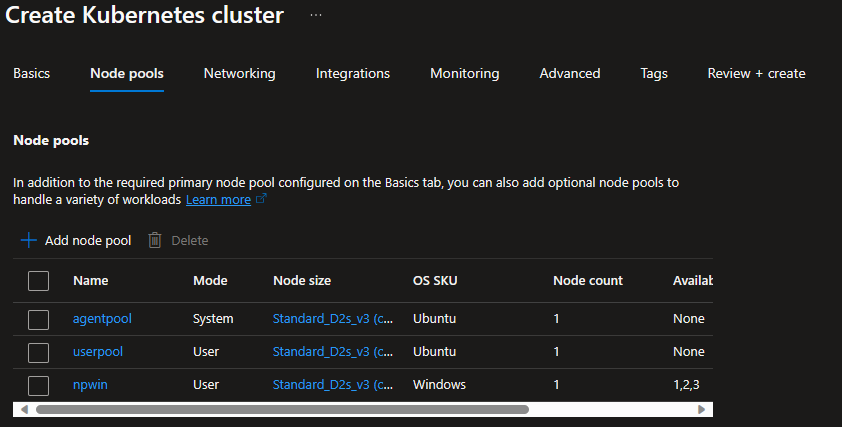


1. Leave all settings on the other tabs set to their defaults and click **Add**.

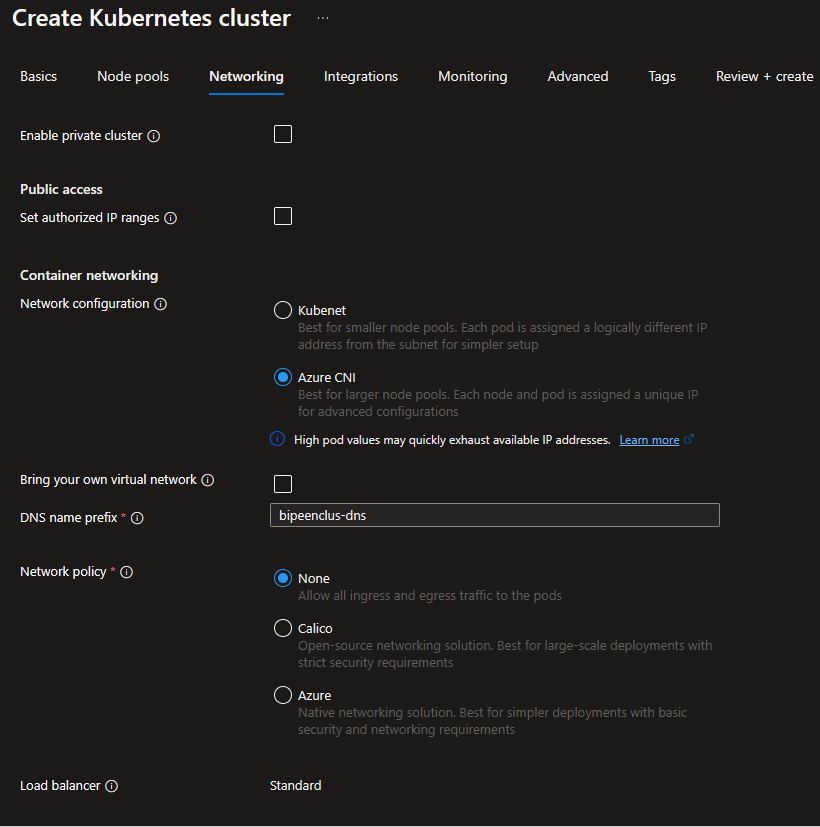
* Change the Node Size for remaining Agentpool and Useepool to D2s\_v3 and Leave the **Scale method** setting set to Manual
* Set the Node Count to 1



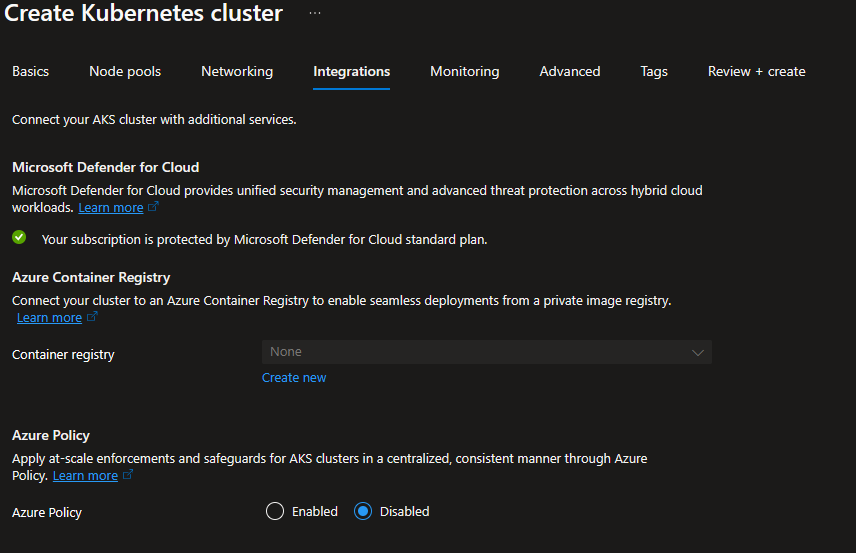
1. This is how the Node pools should look like



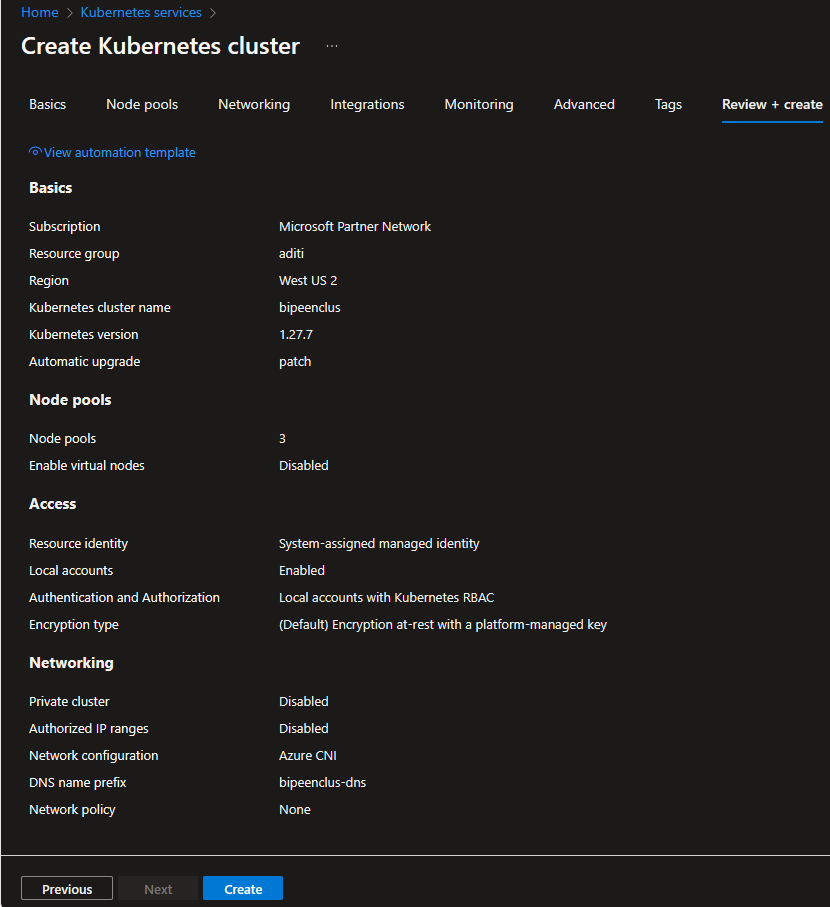
1. Click Next for Networking and select network Policy None



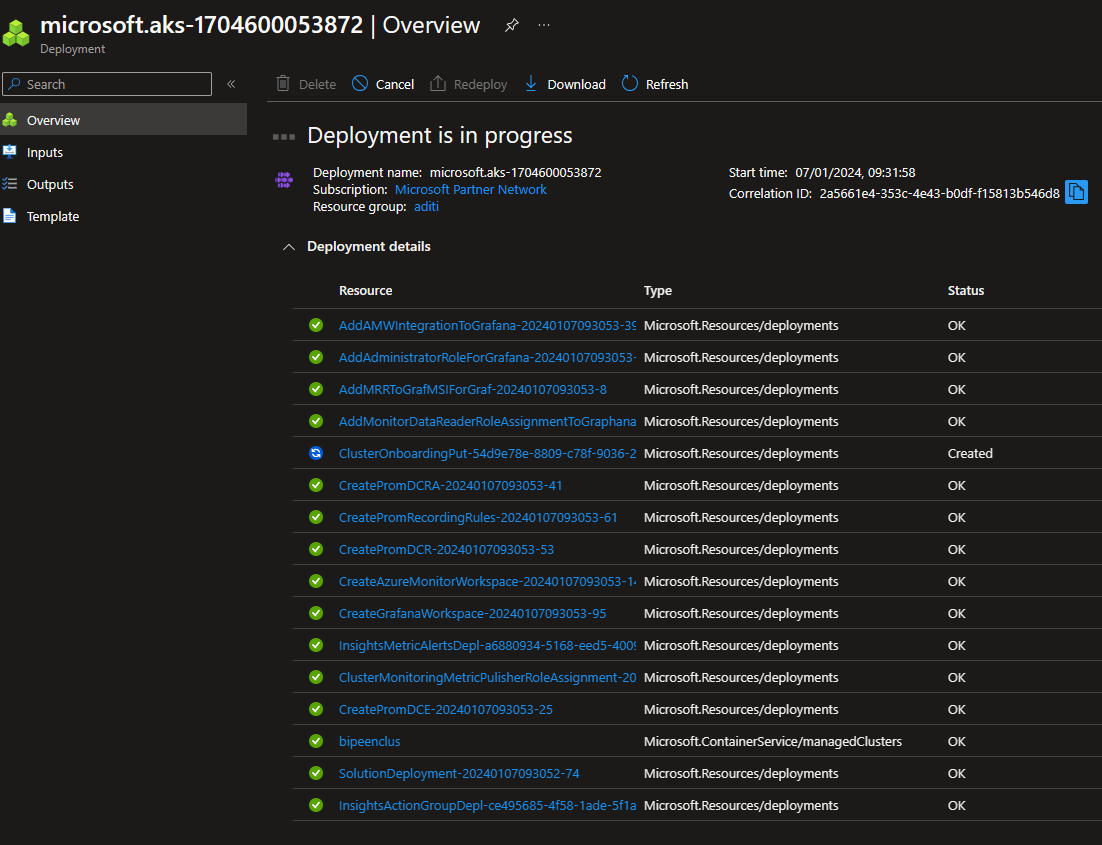
1. Click Next for Integration and Disable Azure Policy



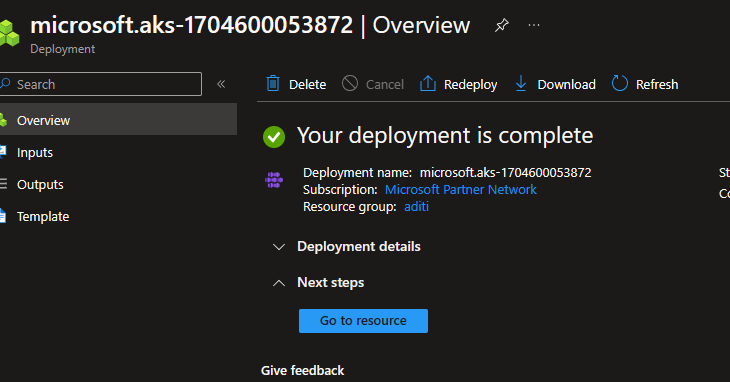
1. Leave all settings on the other tabs set to their defaults.
2. Select **Review + create** to run validation on the cluster configuration.



1. After validation completes, select **Create** to create the AKS cluster
2. It takes a few minutes to create the AKS cluster.



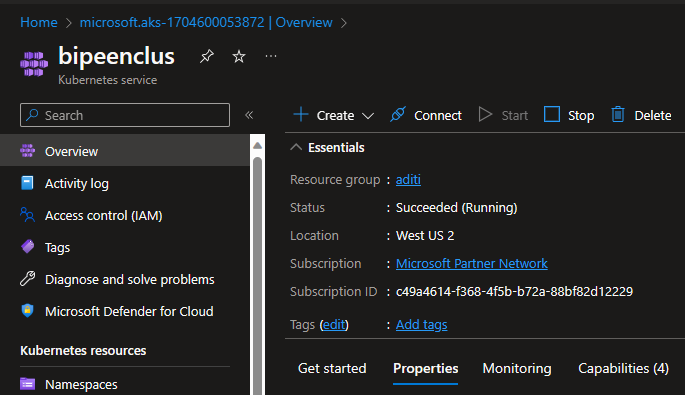
When your deployment is complete, navigate to your resource



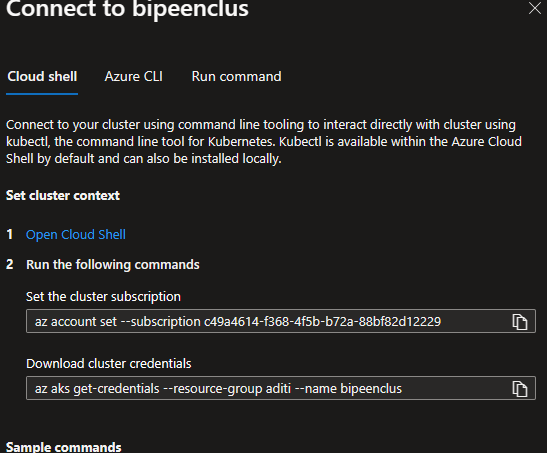
# Connect an AKS cluster

You use [kubectl](https://kubernetes.io/docs/reference/kubectl/), the Kubernetes command-line client, to manage your Kubernetes clusters. kubectl is already installed if you use Azure Cloud Shell

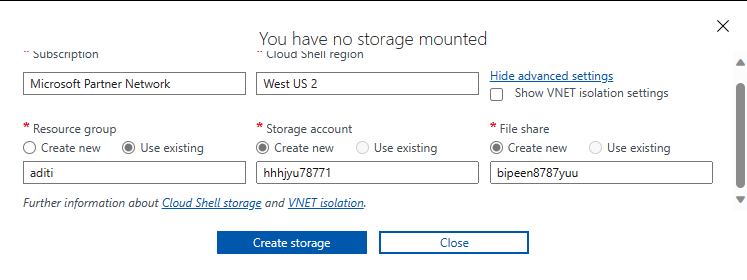
* Click **Connect**



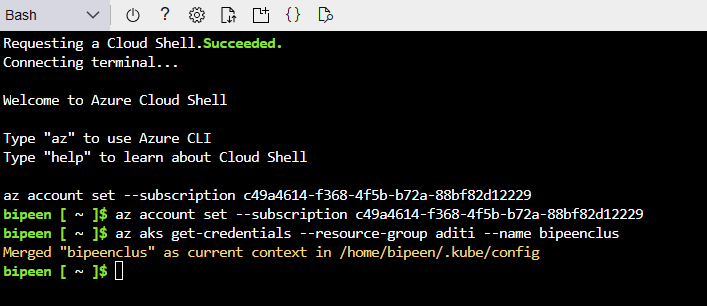
* Click **Open Cloud Shell**



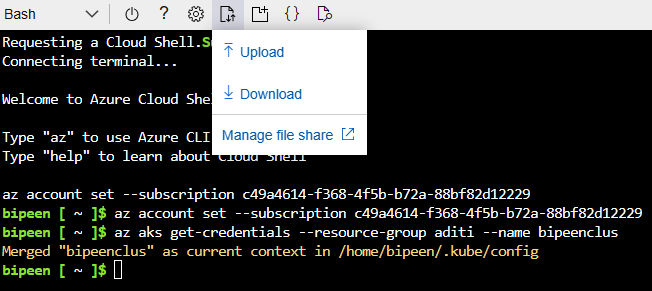
* Set **Region, Resource Group** and Create **new Storage** and **File Share**



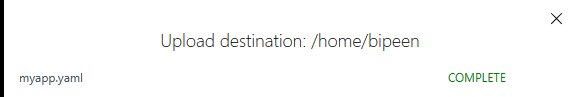
* You should able to see following screen



* Click upload to upload YAML file



* Click upload to upload YAML file



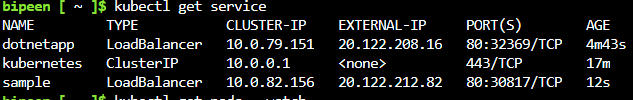
ls



kubectl apply -f sample.yaml



kubectl get service sample –watch



kubectl get pods –watch

