Lab 7 - Secure and govern your cluster with Azure Policies

Introduction

In this Lab you implement governance on your cluster by allowing pods only if they have the required tags so you can easily track the different deployed workloads. In real life you can implement kind of rules such as restricting privileged pods, kind of storage volumes you want to allow etc.

To do that, you will enable (if not already done) Azure Policy Add-on on your cluster, create a `Policy Assignment` to force labels and then you will test it's functioning.

- Expected Lab duration: 30 minutes

Challenge yourself (OPTIONAL)

If you want to challenge yourself, try to perform the following actions on your own:

1. Ensure the Azure Policy Add-On is enabled on your cluster (you can have a look at some related pods in the `kube-system` namespace)
2. Create a `Policy Assignment` of the `Kubernetes cluster pods should use specified labels` policy with your AKS Cluster Resource Group as scope with settings described in the table of Exercise 2:
3. Wait for ~30 minutes/1 hour for the policy to apply
4. Try to create a Pod not following your rule and watch the error message
5. Try to create a Pod following your rule and ensure it's running well

Exercise 1: Ensure Azure Policy Add-On is deployed

To use Azure Policy in AKS, you first need to enable the Add-On. You can do that using the Portal, Azure CLI, PowerShell, Bicep, Terraform or even using Azure Policy itself!

Using Azure Portal

1. On the Azure Portal, go to your cluster resource (you can find it by using its name or by going first to your resource group)
2. Click on **Policies** on the left menu (in the `Settings` part)
3. If Azure Policy Add-On in not enabled, click on `Enable Add-On`, otherwise you can go to next exercise

Using Azure CLI

1. Open **Azure Cloud Shell**

2. Execute the following commands to see if you have the add-on already installed:

   ```
   # Have a look if your cluster already has the add-on installed
   kubectl get deployment -A --show-labels
   ```

3. If you have these deployments you can go to the next exercise:
    - `azure-policy` in `kube-system` namespace
    - `azure-policy-webhook` in `kube-system` namespace
    - `gatekeeper-audit` in `gatekeeper-system` namespace
    - `gatekeeper-controller` in `gatekeeper-system` namespace

4. If you don't have the pods, you will have to deploy the add-on using this command:

```
AKS_CLUSTER="${PREFIX}aks${STUDENT_NB}"
az aks enable-addons --addons azure-policy --name ${AKS_CLUSTER} -g
${RESOURCE_GROUP}
```

Exercise 2: Deploy an Azure Policy to enforce tag governance

To deploy the **Azure Policy** you will use the Azure Portal.

1. On the Azure Portal, go to your cluster resource (you can find it by using its name or by going first to your resource group)
2. Click on **Policies** on the left menu (in the `Settings` part)
3. Click on the link "go to Azure Policy" in the center part (you can also access Azure Policy directly from the search bar)
4. Click on `Assign Policy`
5. Make sure the scope of the Policy is set to your resource group `rg-akstrainingX`
6. Click on the `...` button next to the `Policy Definition`
7. In the search bar, search for `labels` or copy paste directly `Kubernetes cluster pods should use specified labels`
8. Make sure the Policy enforcement is `Enabled`
9. Click on `Next`
10. On the **Advanced** page, click on `Next`
11. On the **Parameter** page, **un**check the `Only show parameters that need input or review` then enter these parameters:

Setting	Value
Effect	Deny
Namespace exclusions	no change
Namespace inclusions	`["policylab"]`
Kubernetes label selector	no change
List of labels	`["app"]`

12. This will deny pod creation if the `app` label is not present for all pods in the `policylab` namespace
13. Click on `Review + Create`
14. Click on `Create`

The policy deployment will take ~30minutes. Come back later to test your policy.

Exercise 3: Test your policy

In tis exercise you will ensure your policy has been successfully deployed and you will test it by creating compliant and non-compliant pods.

1. Open **Azure Cloud Shell**
2. Execute the following command to verify your policy has been successfully deployed:

   ```
   kubectl get constrainttemplates k8sazurev1podenforcelabels
   ```

3. If you have a result you can continue, otherwise wait a few more minutes
4. The policy will apply only in the `policylab` namespace (to avoid any issues with other labs). This namespace doesn't exist yet, so let's create it:

   ```
   kubectl create ns policylab
   ```

5. Try to create a pod in this namespace, you should have an error message telling you need to have the `app` label

   ```
   kubectl run nginx --image nginx:latest -n policylab
   ```

6. You can have more info about the `constraint` in your AKS cluster by looking at these resources:

   ```
   kubectl describe k8sazurev1podenforcelabels.constraints.gatekeeper.sh
   ```

7. Try now to create a pod with the corresponding label

   ```
   kubectl run nginx --image nginx:latest -n policylab --labels app=nginx
   kubectl get pod -n policylab --show-labels
   ```

8. You can now clean the environment

   ```
   kubectl delete ns policylab
   az policy assignment delete -g "rg-akstraining${STUDENT_NB}" -n "Kubernetes cluster pods should use specified labels"
   ```