

**Lab Manual- Azure Kubernetes Services Add-ons and Extensions**

**Prepared for**:

**Date:** 18th Dec 2023

**Prepared by:**

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

**Contributor:**

Contents

[1. Objective 3](#_Toc157022559)

[1. Add-ons​ 3](#_Toc157022560)

[2. Extensions 4](#_Toc157022561)

[1. Difference between extensions and add-ons 4](#_Toc157022562)

[2. Create 3 Node Azure Kubernetes Cluster 4](#_Toc157022563)

[3. Create log alerts from Container insights 12](#_Toc157022564)

[4. Syslog collection with Container Insights 24](#_Toc157022565)

[1. From the Azure portal 24](#_Toc157022566)

[2. Using Azure CLI commands 26](#_Toc157022567)

[1. How to access Syslog data 26](#_Toc157022568)

[1. Access using built-in workbooks 26](#_Toc157022569)

[1. Editing your Syslog collection settings 32](#_Toc157022570)

# Objective

## Add-ons[​](https://azure.github.io/Cloud-Native/cnny-2023/aks-extensions-addons/#add-ons)

Add-ons provide a supported way to extend AKS. Installation, configuration and lifecycle are managed by AKS following [pre-determine updates rules](https://learn.microsoft.com/azure/aks/integrations?WT.mc_id=containers-84290-joarteir#add-ons).

As an example, let's enable Container Insights with the monitoring addon. on an existing AKS cluster using **az aks enable-addons --addons** CLI command

**az aks enable-addons \  
 --name MyManagedCluster \  
 --resource-group MyResourceGroup \  
 --addons monitoring**

or you can use **az aks create --enable-addons** when creating new clusters

**az aks create \  
 --name MyManagedCluster \  
 --resource-group MyResourceGroup \  
 --enable-addons monitoring**

The current available add-ons are:

1. **http\_application\_routing** - Configure ingress with automatic public DNS name creation. Only recommended for development.
2. **monitoring** - Container Insights monitoring.
3. **virtual-node** - CNCF virtual nodes open source project.
4. **azure-policy** - Azure Policy for AKS.
5. **ingress-appgw** - Application Gateway Ingress Controller (AGIC).
6. **open-service-mesh** - CNCF Open Service Mesh project.
7. **azure-keyvault-secrets-provider** - Azure Key Vault Secrets Provider for Secret Store CSI Driver.
8. **web\_application\_routing** - Managed NGINX ingress Controller.
9. **keda** - CNCF Event-driven autoscaling project.

For more details, get the updated list below

<https://learn.microsoft.com/en-gb/azure/aks/integrations?WT.mc_id=containers-84290-joarteir#available-add-ons>

## Extensions

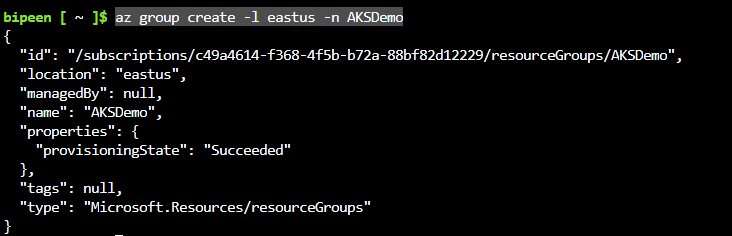
Cluster extensions build on top of certain Helm charts and provide an Azure Resource Manager-driven experience for installation and lifecycle management of different Azure capabilities on top of your Kubernetes cluster.

### Difference between extensions and add-ons

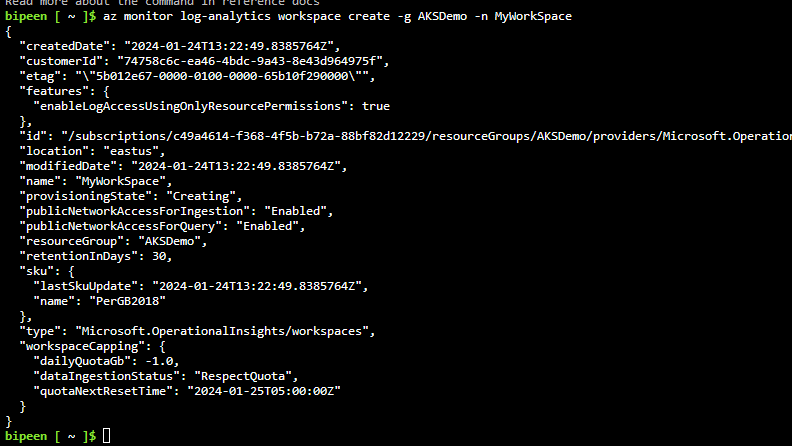
Extensions and add-ons are both supported ways to add functionality to your AKS cluster. When you install an add-on, the functionality is added as part of the AKS resource provider in the Azure API. When you install an extension, the functionality is added as part of a separate resource provider in the Azure API.

# Create 3 Node Azure Kubernetes Cluster

az group create -l eastus -n AKSDemo

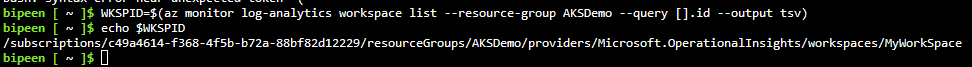


az monitor log-analytics workspace create -g AKSDemo -n MyWorkSpace



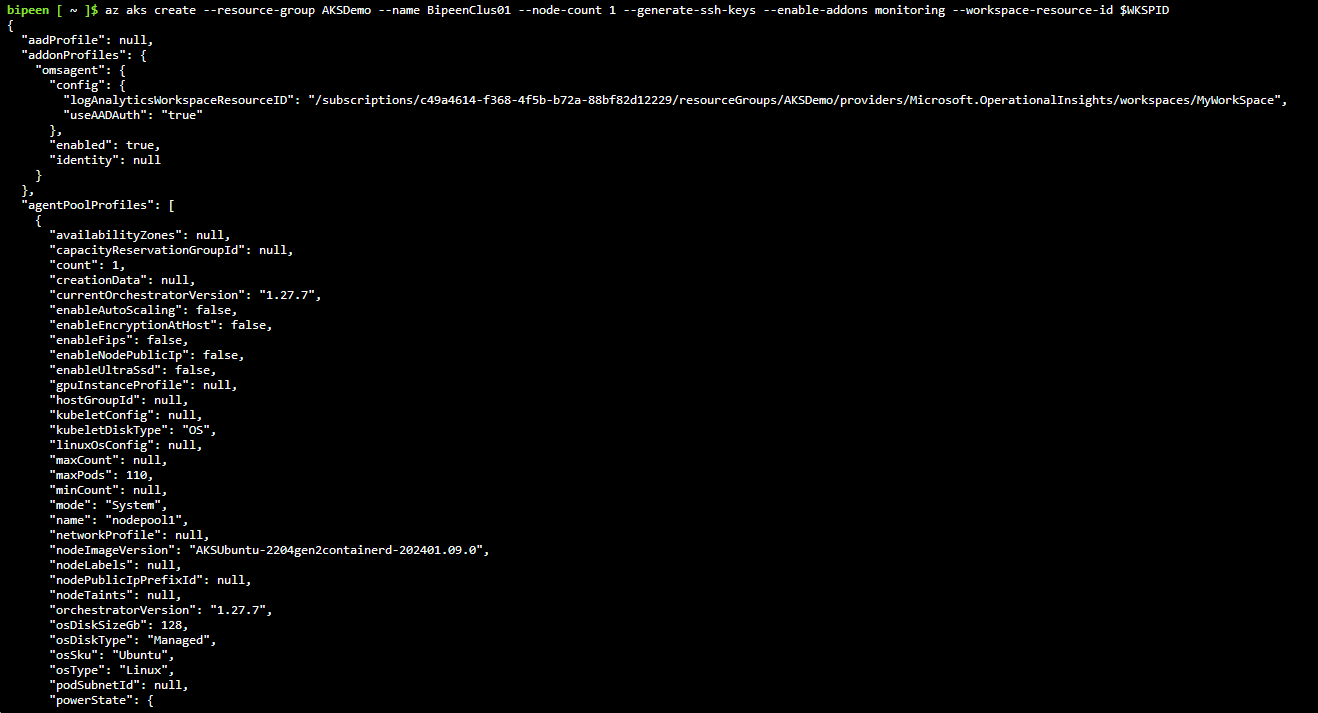
WKSPID=$(az monitor log-analytics workspace list --resource-group AKSDemo --query [].id --output tsv)

echo $WKSPID



To create an AKS cluster, use the [az aks create](https://learn.microsoft.com/en-us/cli/azure/aks" \l "az-aks-create) command. The following example creates a cluster named BipeenClus01 with one node and generate SSH-key. It will automatically use 3 Nodes when you don’t specify node count

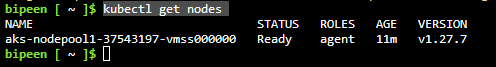
az aks create --resource-group AKSDemo --name BipeenClus01 --node-count 1 --generate-ssh-keys --enable-addons monitoring --workspace-resource-id $WKSPID



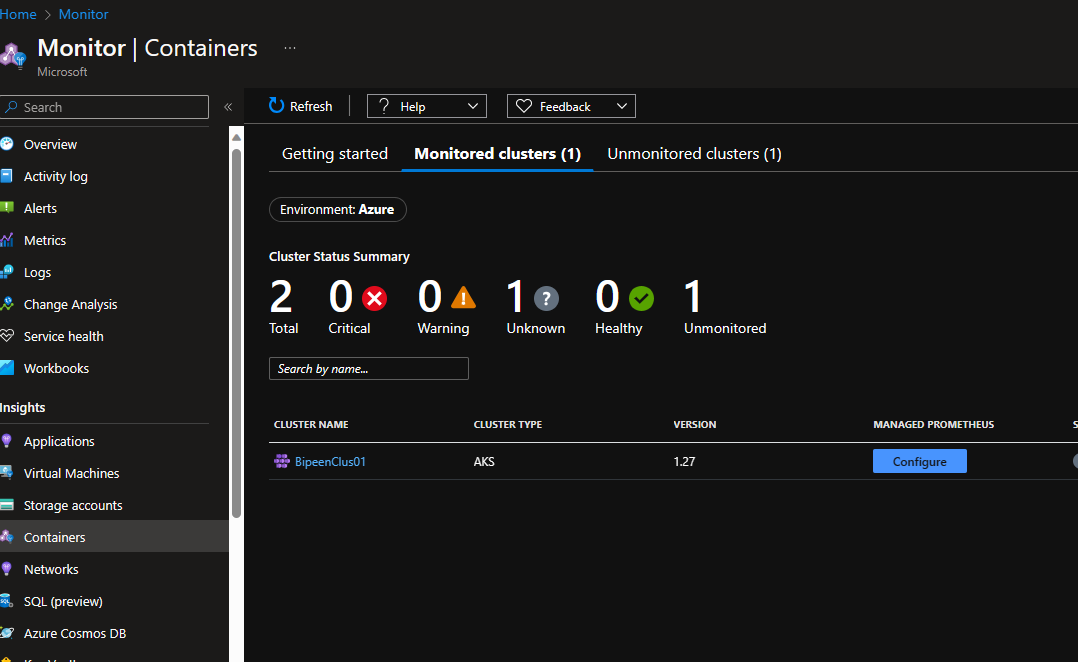
az aks get-credentials --resource-group AKSDemo --name BipeenClus01 --overwrite-existing



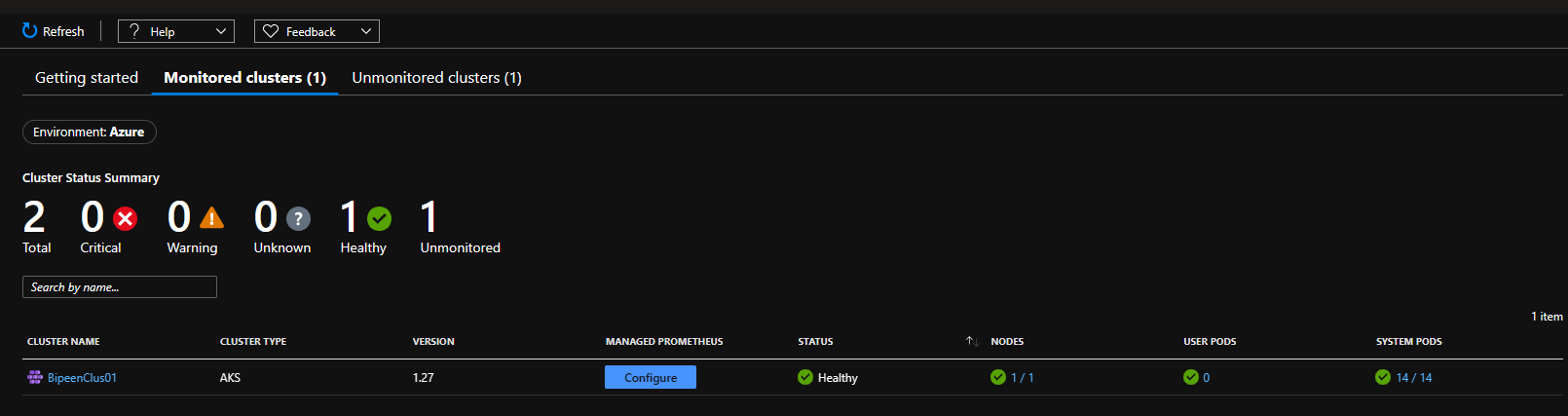
kubectl get nodes



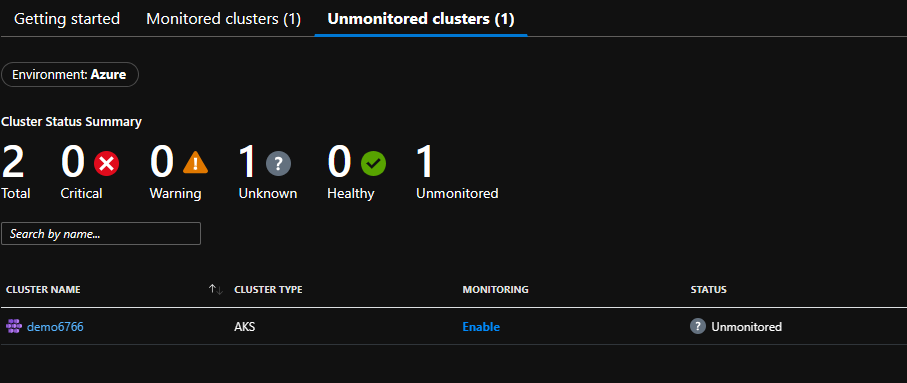
Now in portal search Monitor and inside monitor cleikc container🡪 Monitored cluster



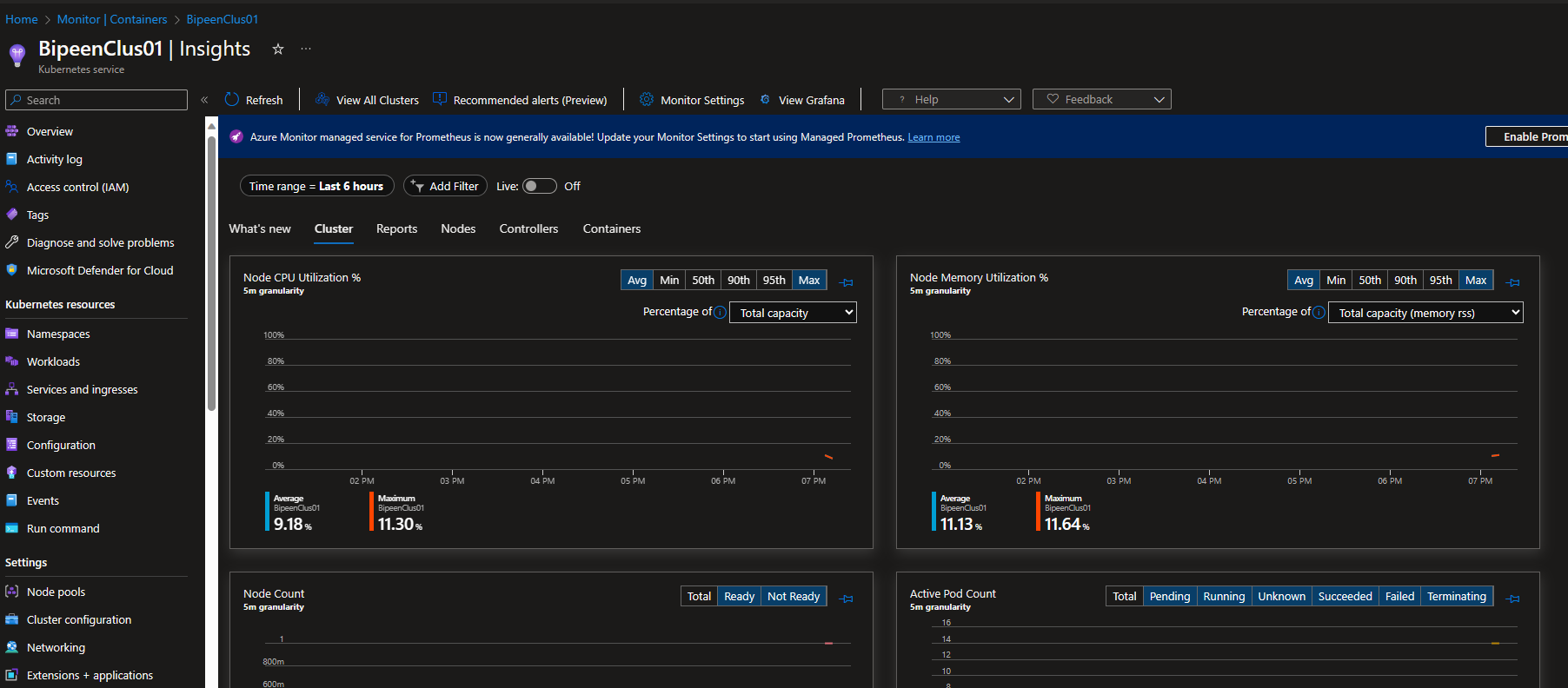
Wait for 4-5 min and refresh the page you should see the status



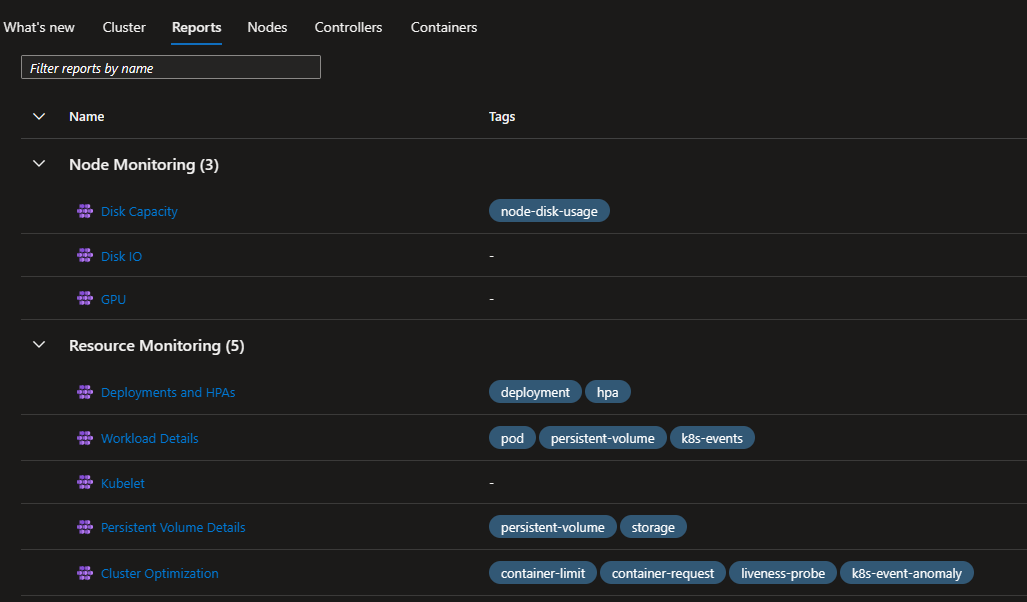
If you have any other clustered which is not monritor you have change to enable it



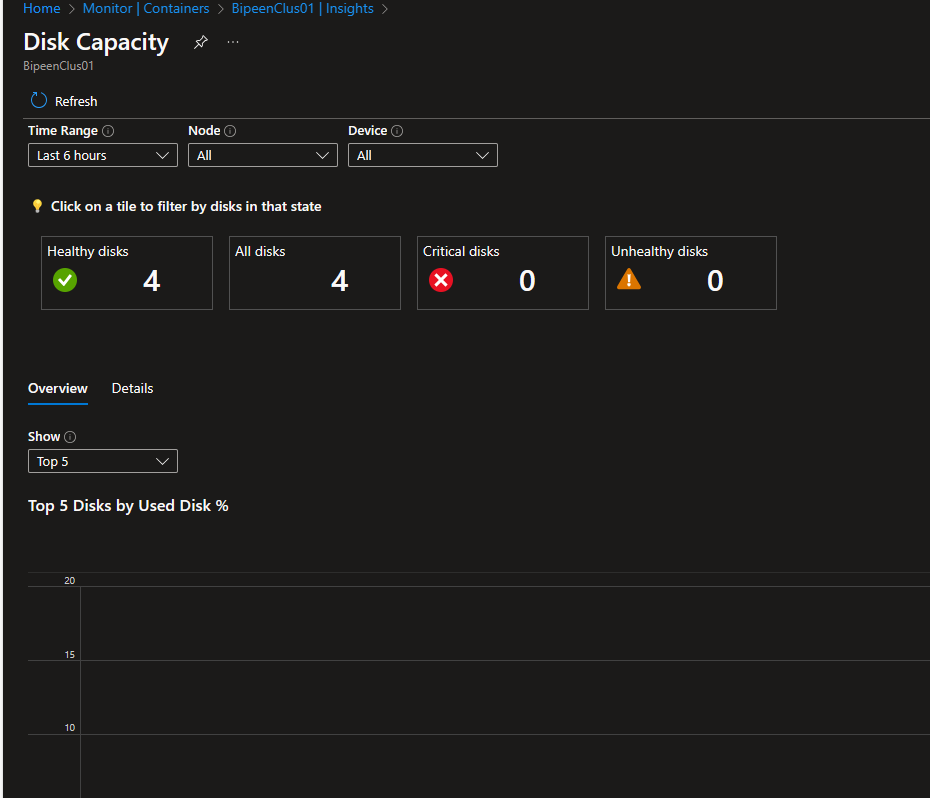
Click BipeenClus1



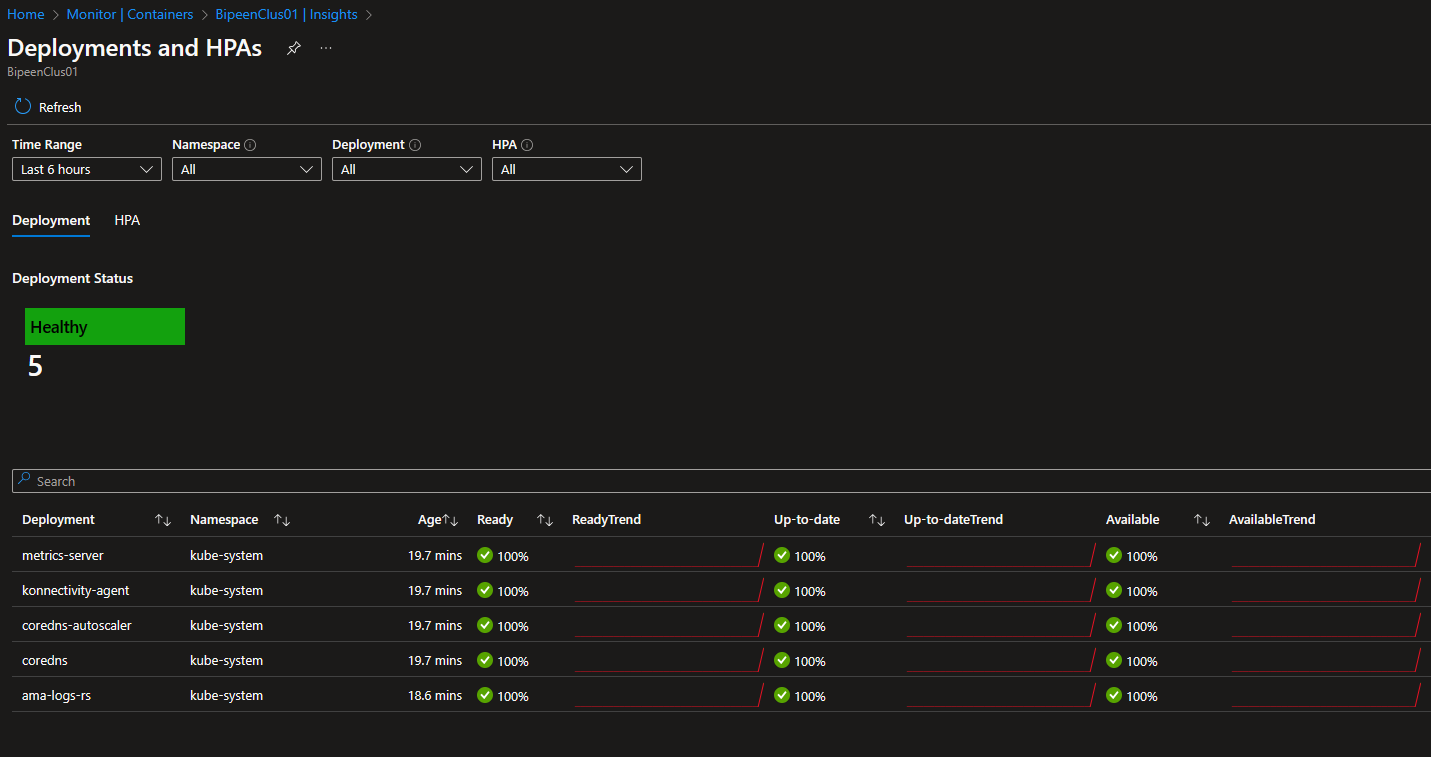
Report Tab



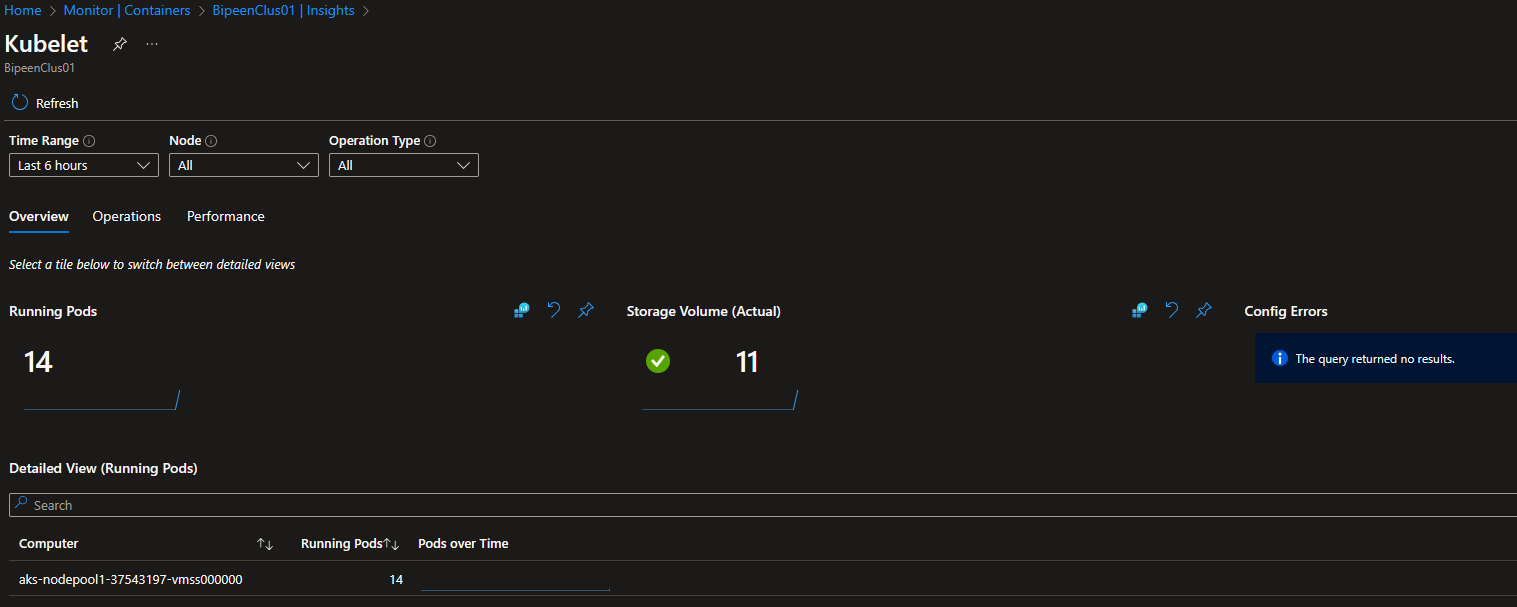
Disk Capacity report



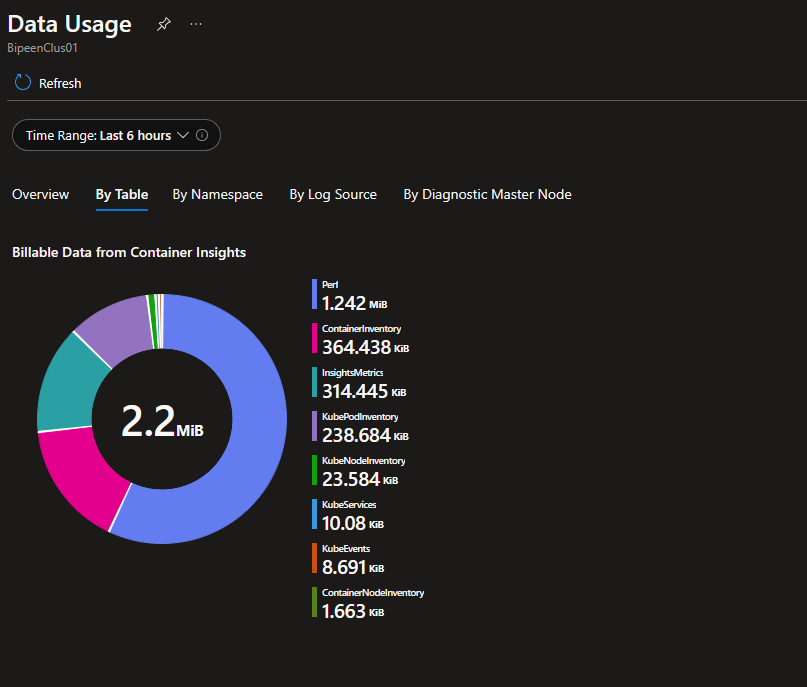
Deployment Report



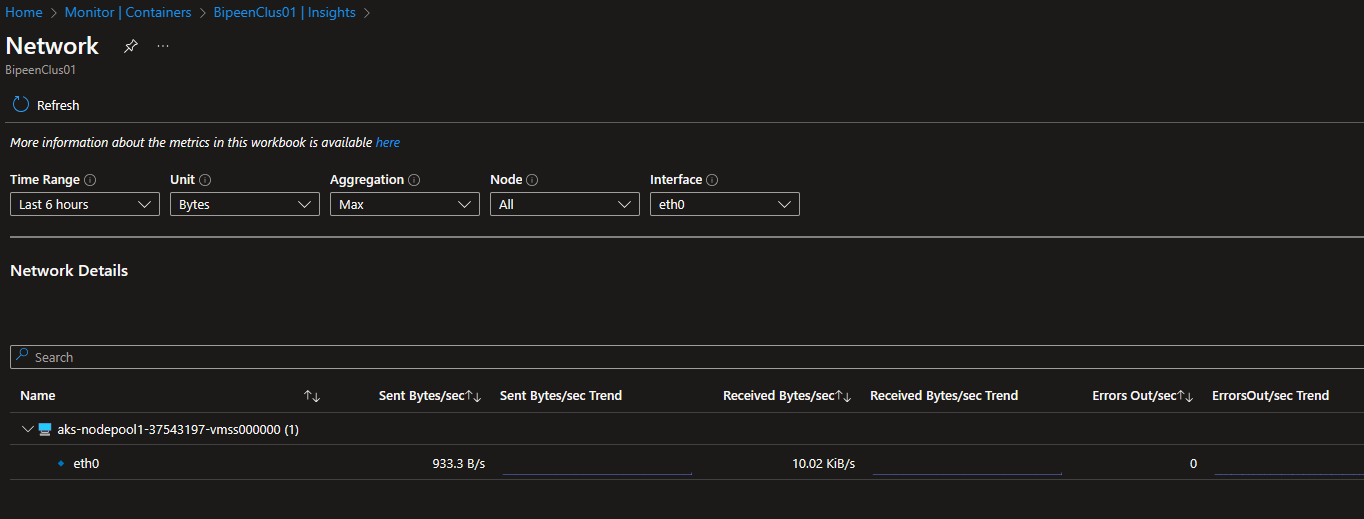
Kubelet Report



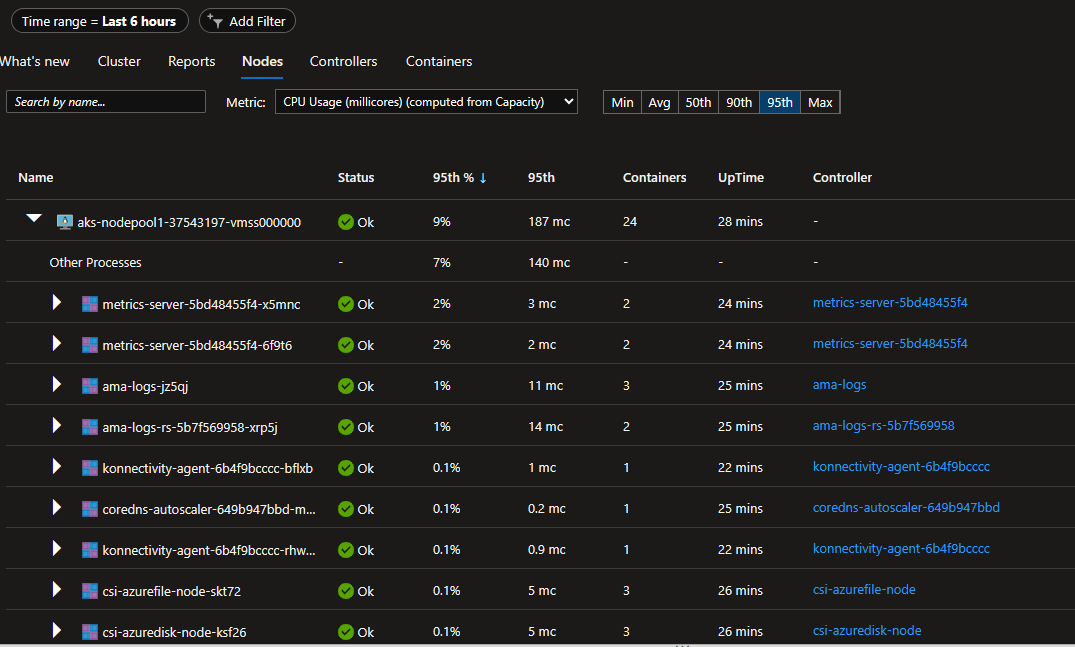
Data Usage by Table



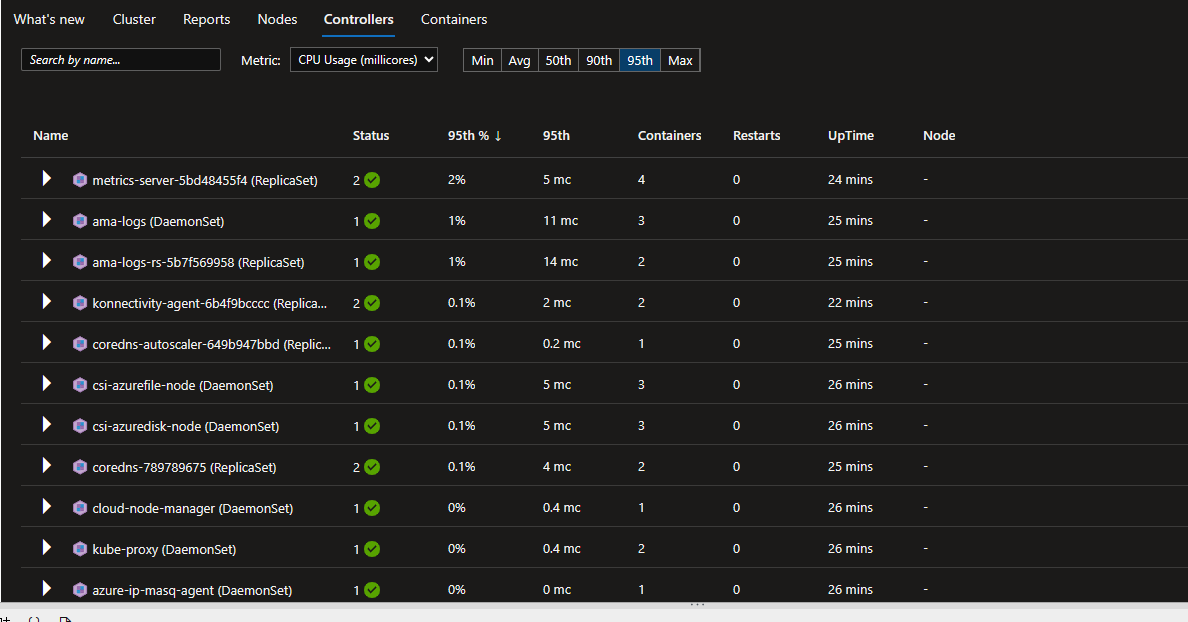
Network Report



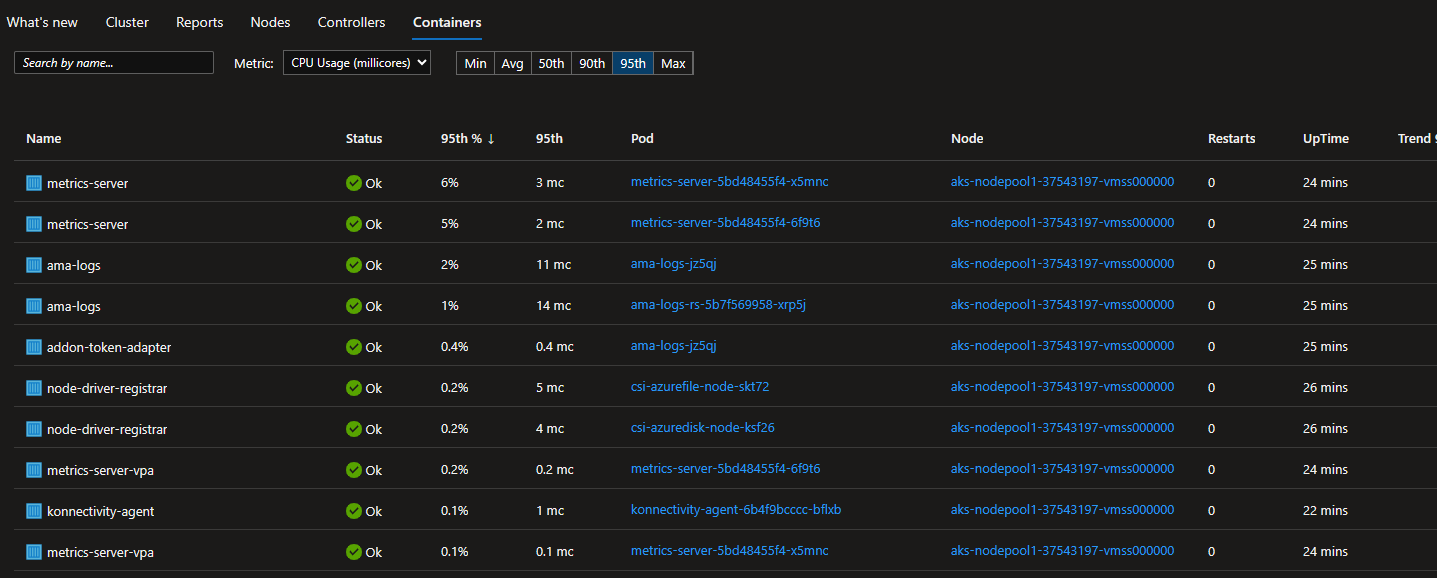
Nodes Report



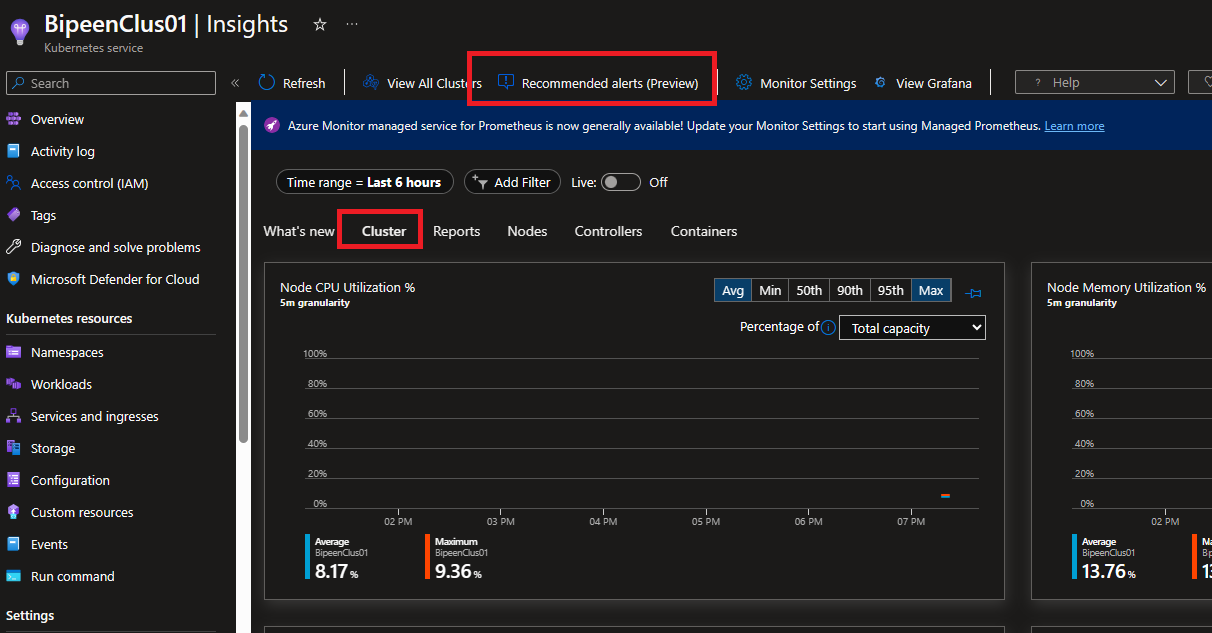
Controller Report



Containerts Report



Now Click back cluster and then upper menue Recommended Alerts(Preview)

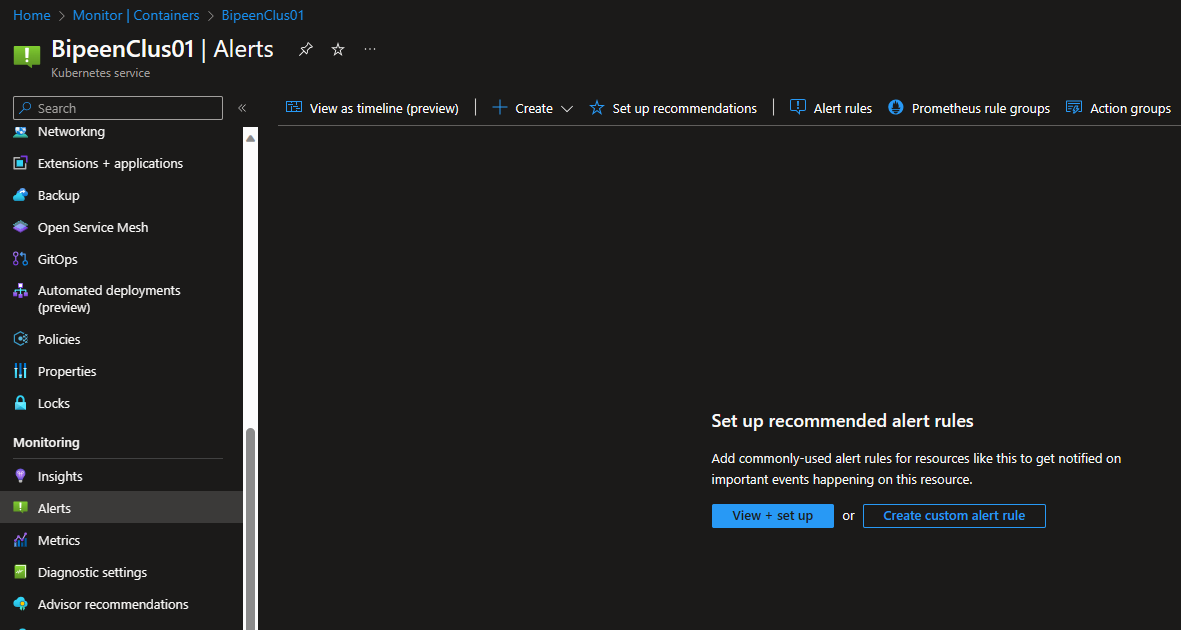


# Create log alerts from Container insights

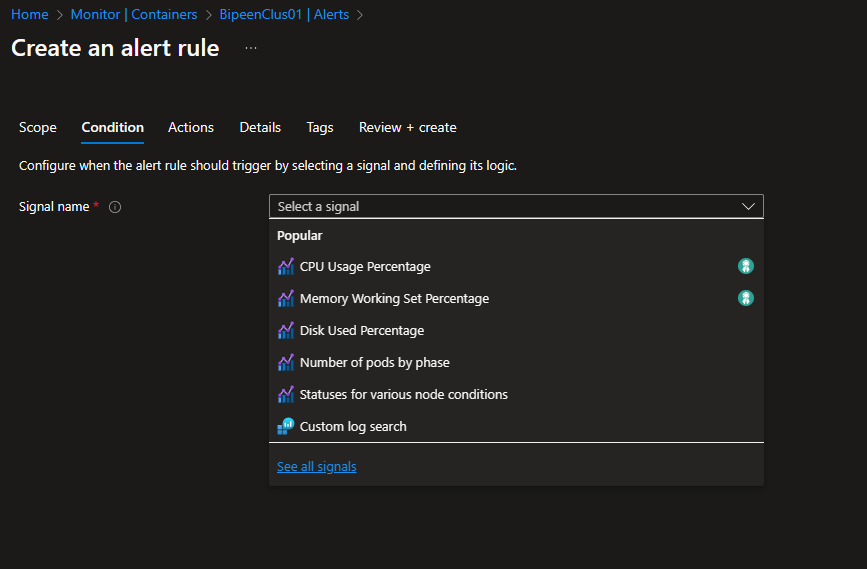
Container insights monitors the performance of container workloads that are deployed to managed or self-managed Kubernetes clusters. To alert on what matters, this article describes how to create log-based alerts for the following situations with Azure Kubernetes Service (AKS) clusters:

* When CPU or memory utilization on cluster nodes exceeds a threshold
* When CPU or memory utilization on any container within a controller exceeds a threshold as compared to a limit that's set on the corresponding resource
* NotReady status node counts
* Failed, Pending, Unknown, Running, or Succeeded pod-phase counts
* When free disk space on cluster nodes exceeds a threshold

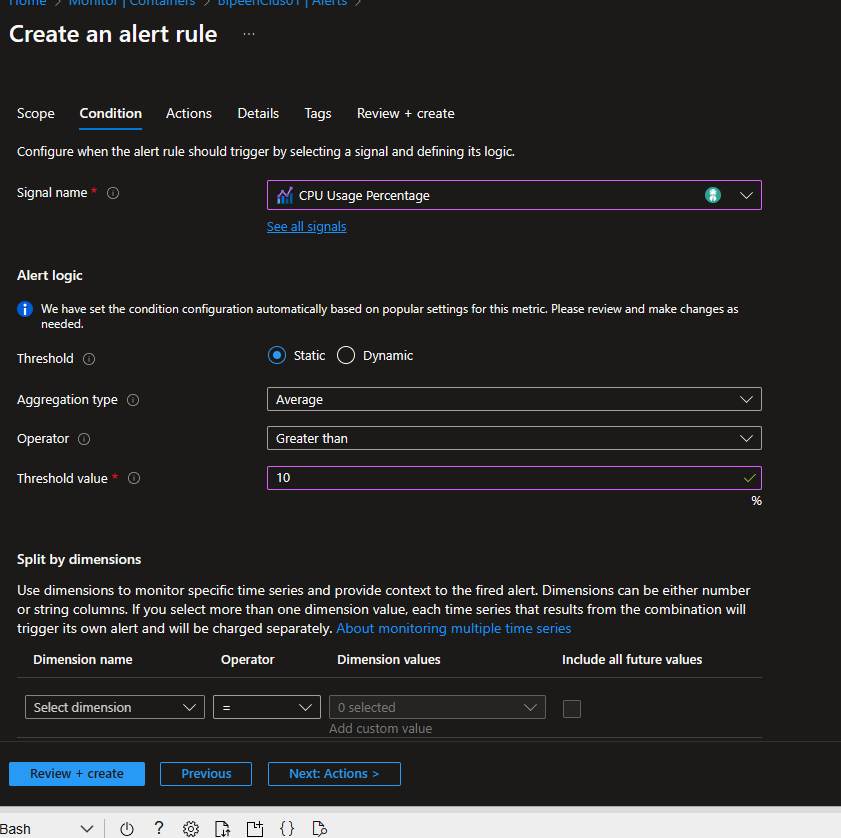
To alert for high CPU or memory utilization, or low free disk space on cluster nodes, use the queries that are provided to create a metric alert or a metric measurement alert. Metric alerts have lower latency than log alerts, but log alerts provide advanced querying and greater sophistication. Log alert queries compare a datetime to the present by using the now operator and going back one hour. (Container insights stores all dates in Coordinated Universal Time [UTC] format.



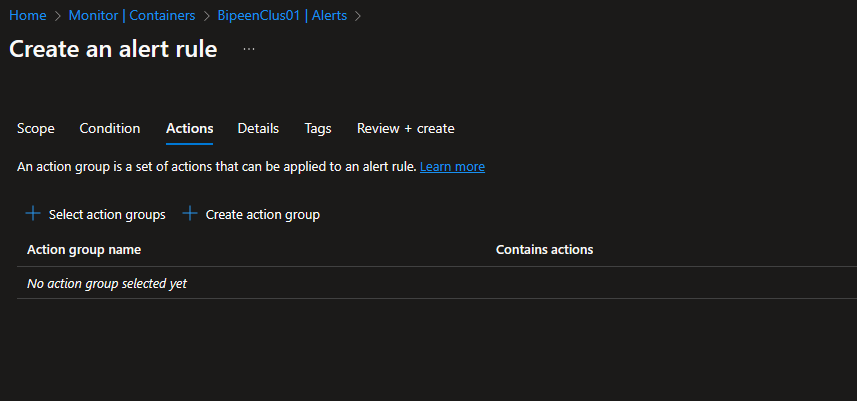
**Select CPU Percentage**



Spevcify value greater than 10

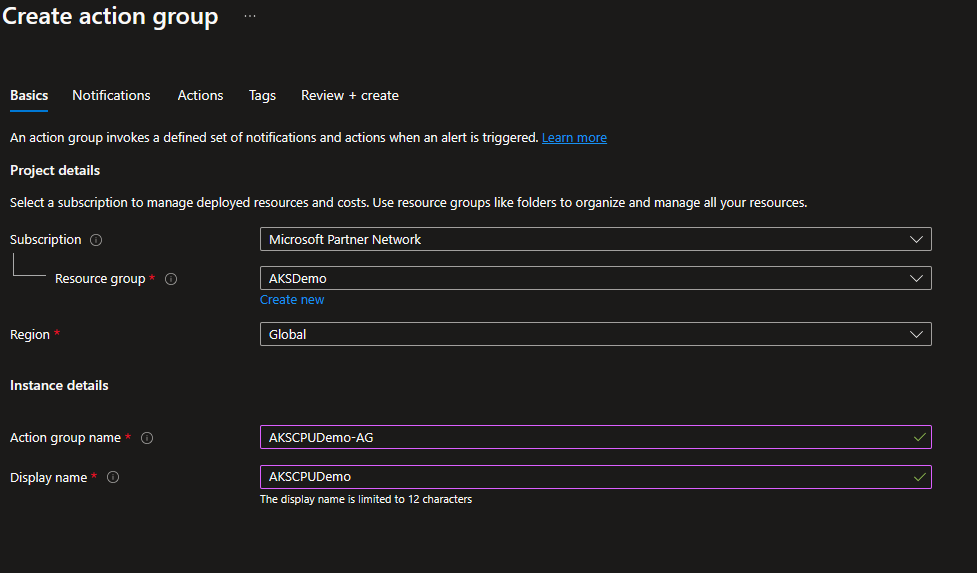


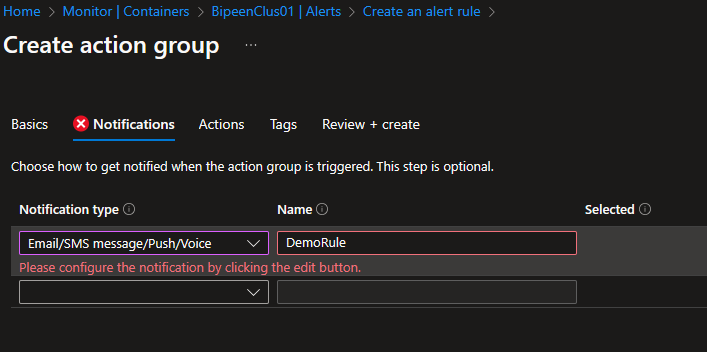
Next create Action group

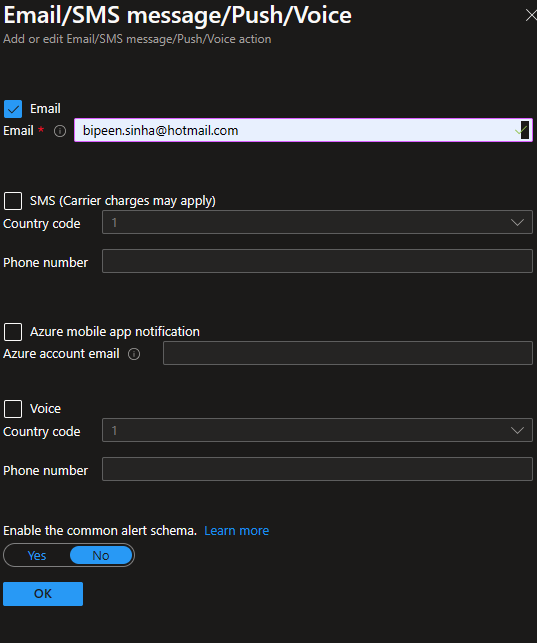


Action GroupName = AKSCPUDemo-AG

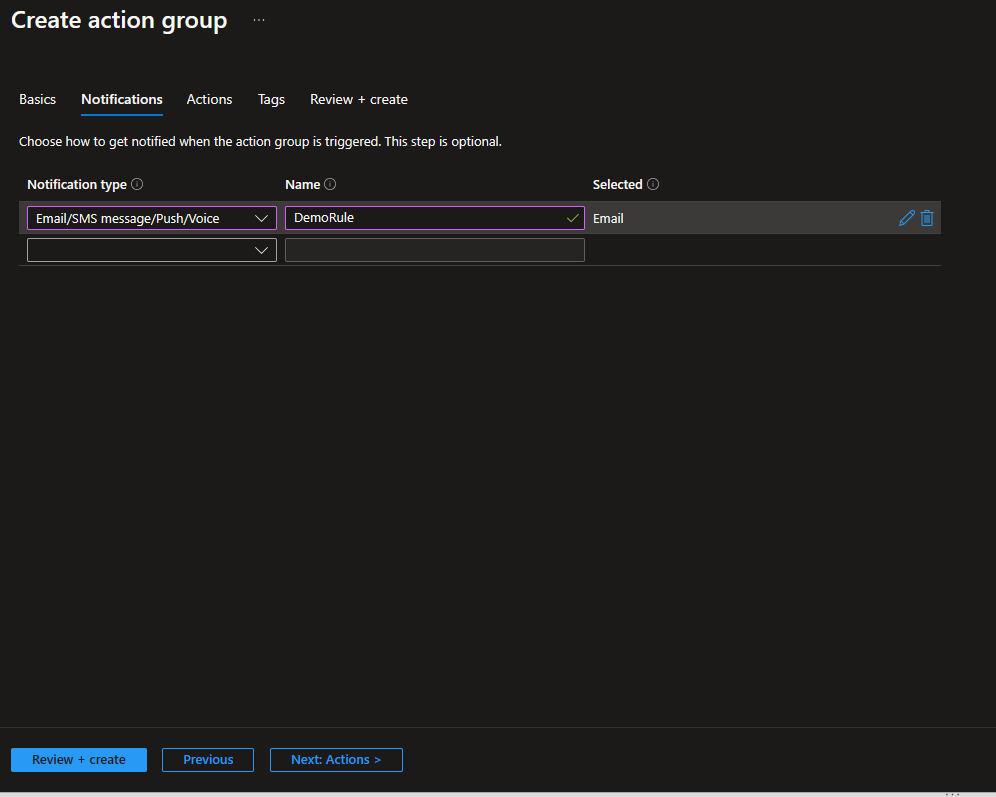
DisplayName = AKSCPUDemo



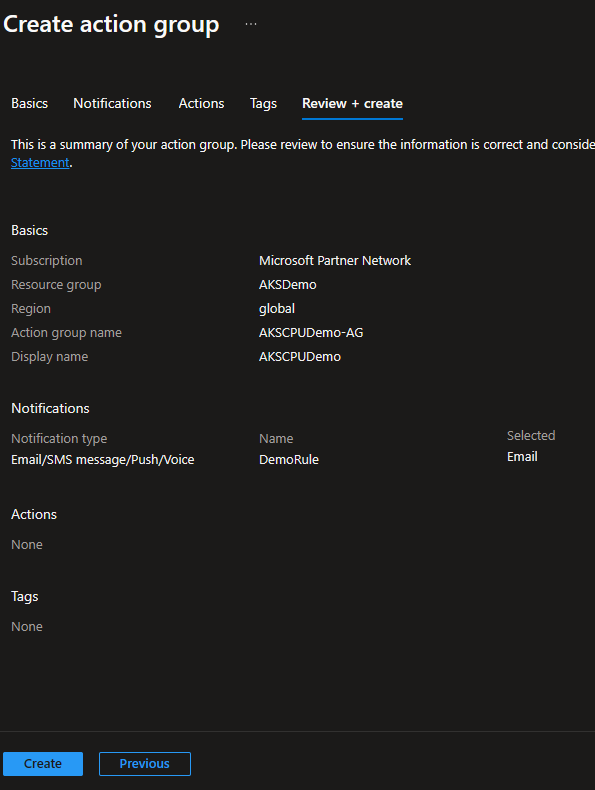




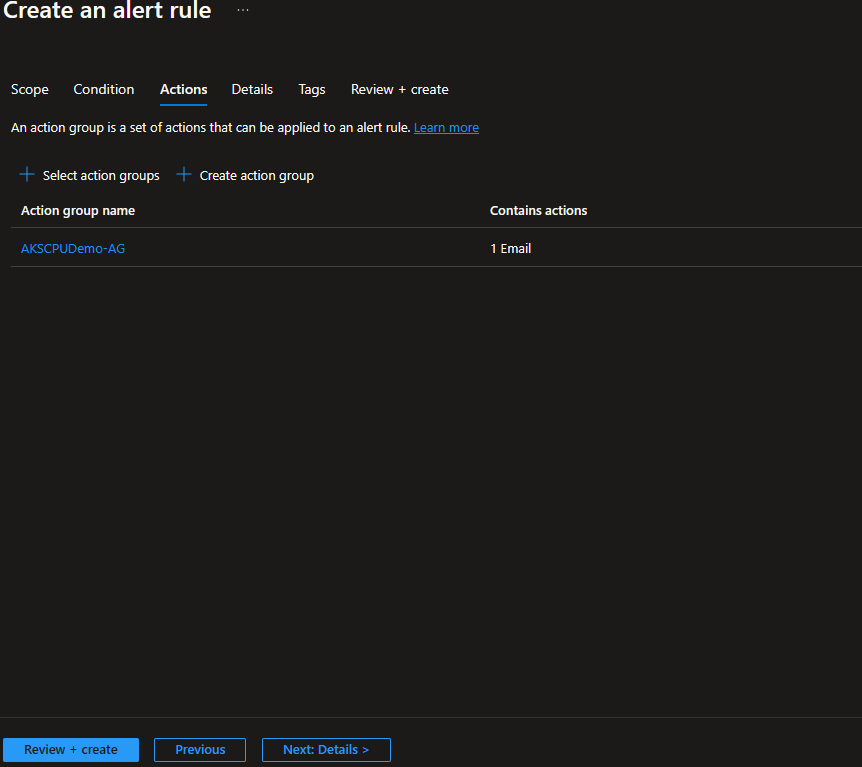
Clcik ok to Save

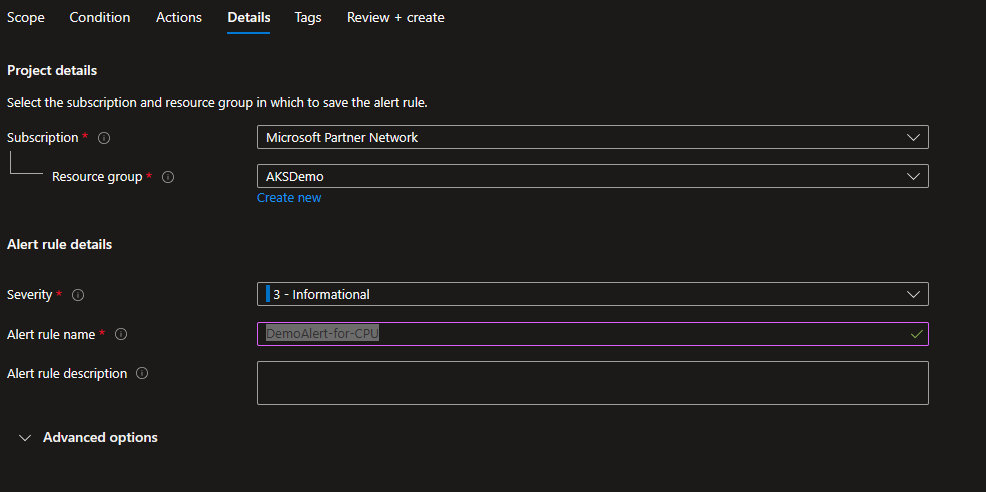


Click Review and Create’

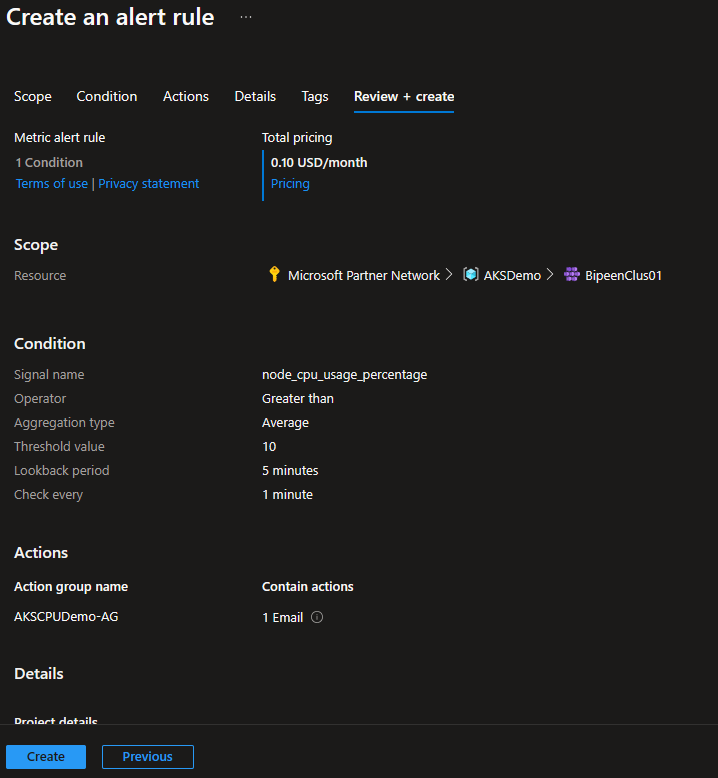


Now Click Next to Details and Set Alert Rule Name = DemoAlert-for-CPU





Now Create the Alert Rulke



Cpu.yaml

apiVersion: v1

kind: Pod

metadata:

  name: cpu-demo

spec:

  containers:

  - image: cpu-demo-ctr

    name: vish/stress

    resources:

      requests:

        cpu: "0.5"

      limits:

        cpu: "1"

    args:

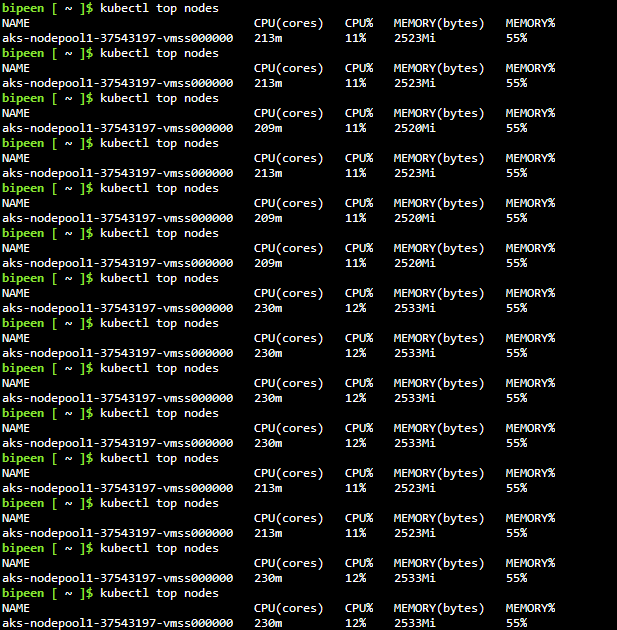
    - -cpu

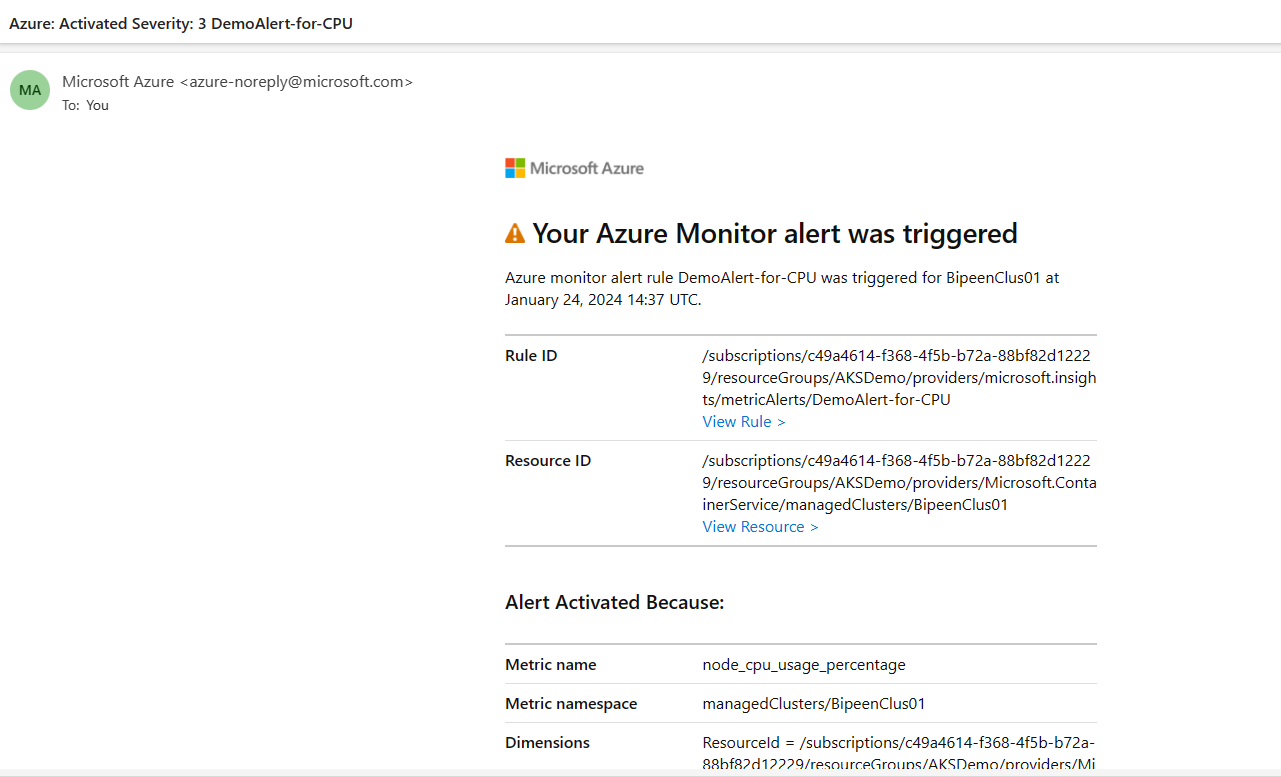
    - "4"

kubectl apply -f cpu.yaml

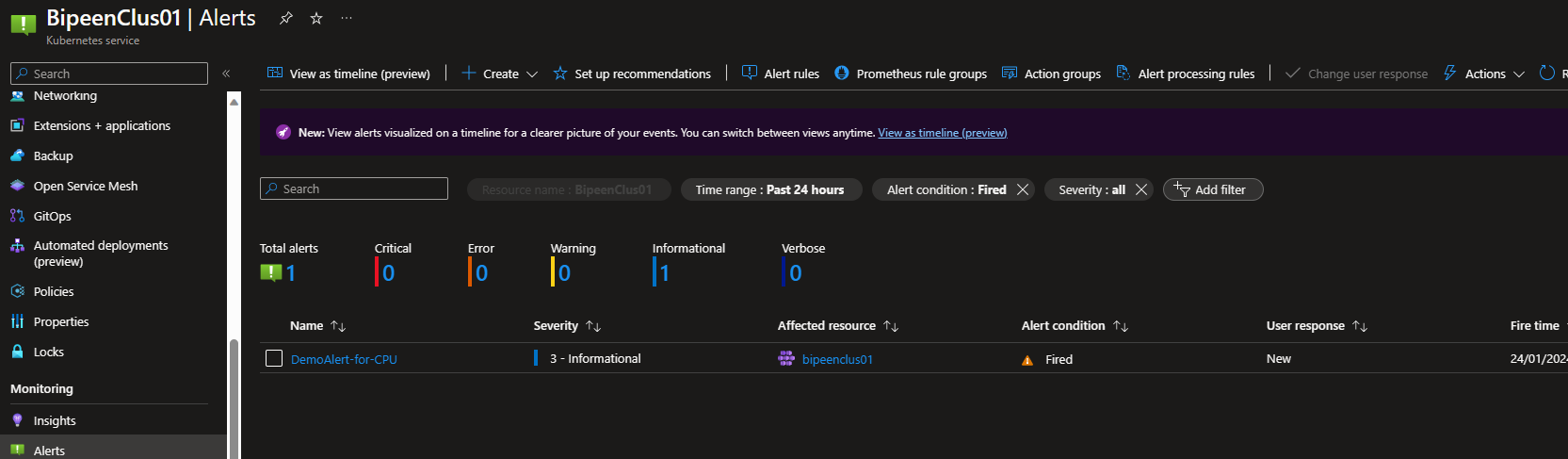


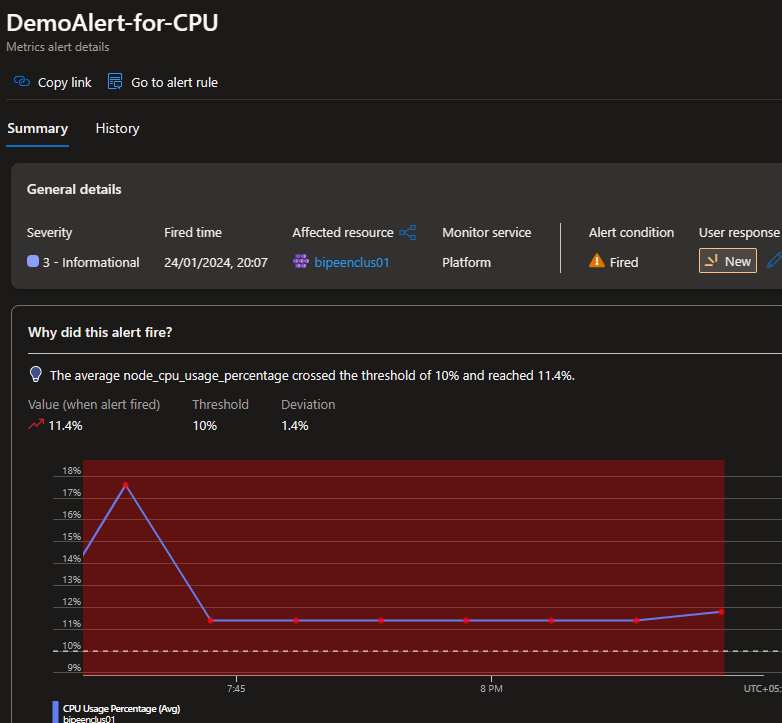
kubectl top nodes



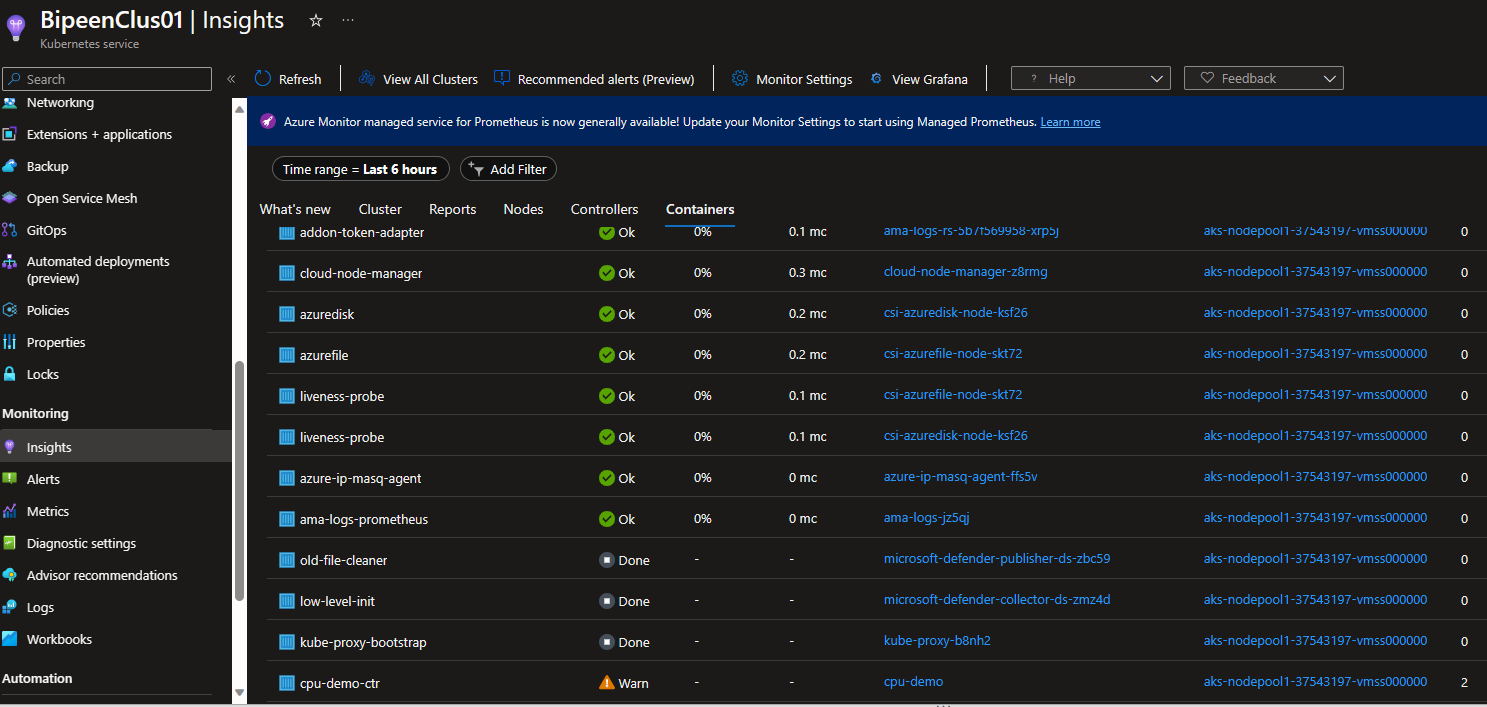


You can see this in Alret





Yiu can also see it in container



Now delete the po

kubectl delete po cpu-demo

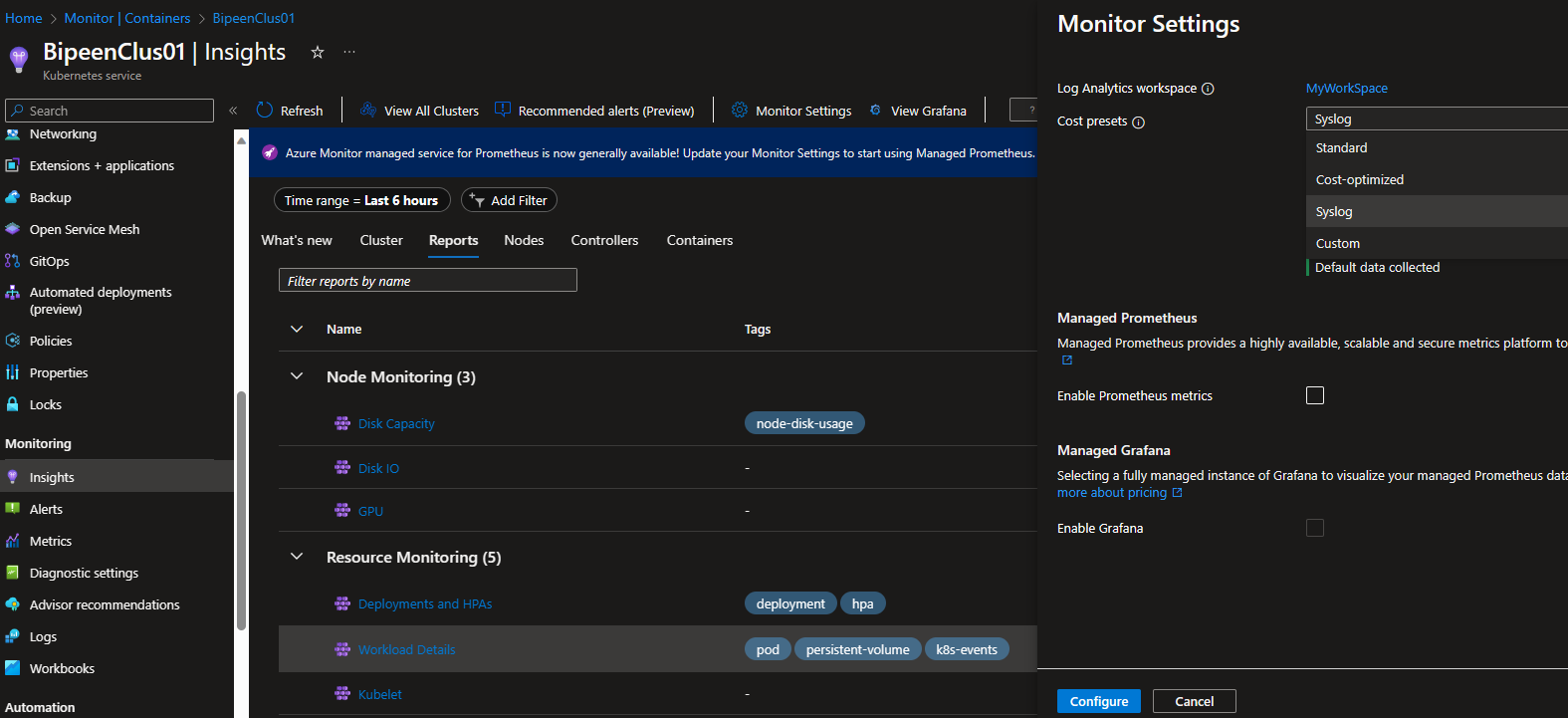


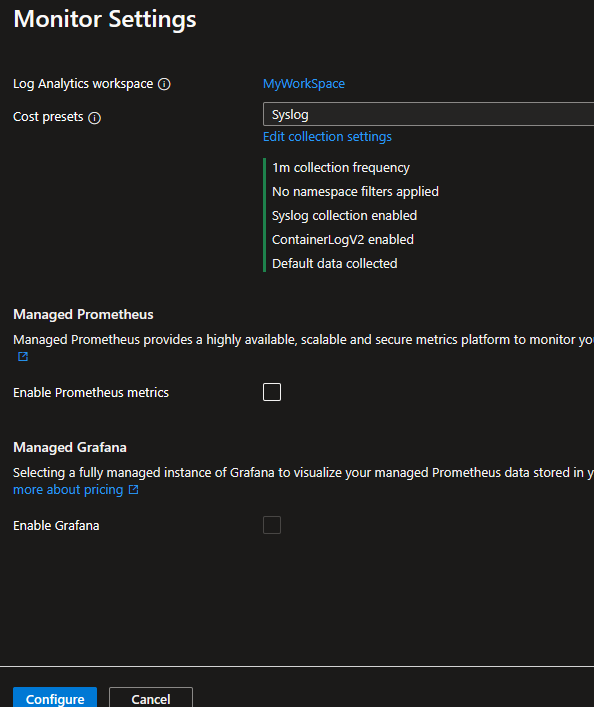
# Syslog collection with Container Insights

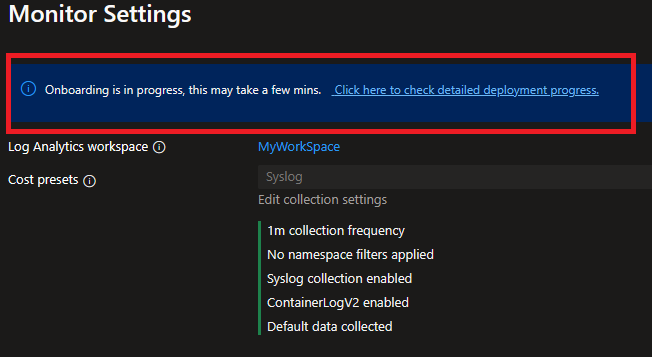
Container Insights offers the ability to collect Syslog events from Linux nodes in your [Azure Kubernetes Service (AKS)](https://learn.microsoft.com/en-gb/azure/aks/intro-kubernetes) clusters. This includes the ability to collect logs from control plane components like kubelet. Customers can also use Syslog for monitoring security and health events, typically by ingesting syslog into a SIEM system like [Microsoft Sentinel](https://azure.microsoft.com/products/microsoft-sentinel/#overview).

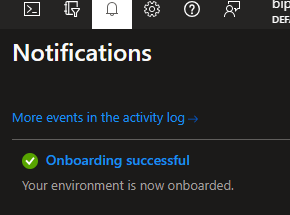
### From the Azure portal

Navigate to your cluster. Open the Insights tab for your cluster. Open the Monitor Settings panel. Click on Edit collection settings, then check the box for Enable Syslog collection









### Using Azure CLI commands

Use the following command in Azure CLI to enable syslog collection when you create a new AKS cluster.

az aks create -g syslog-rg -n new-cluster --enable-managed-identity --node-count 1 --enable-addons monitoring --enable-msi-auth-for-monitoring --enable-syslog --generate-ssh-key

Use the following command in Azure CLI to enable syslog collection on an existing AKS cluster.

az aks enable-addons -a monitoring --enable-msi-auth-for-monitoring --enable-syslog -g syslog-rg -n existing-cluster

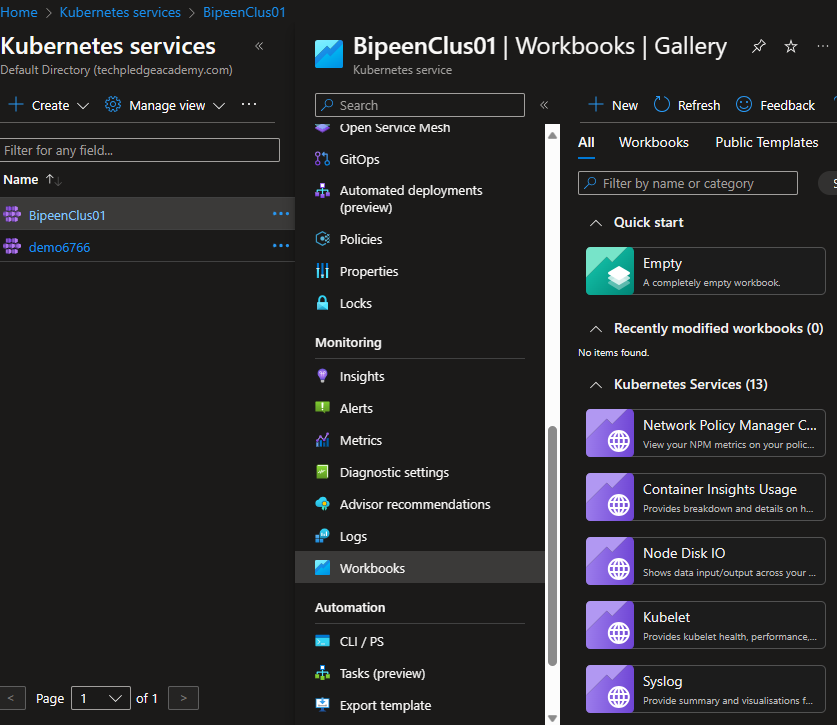
## How to access Syslog data

Once enabled it may take 25-60 Minute

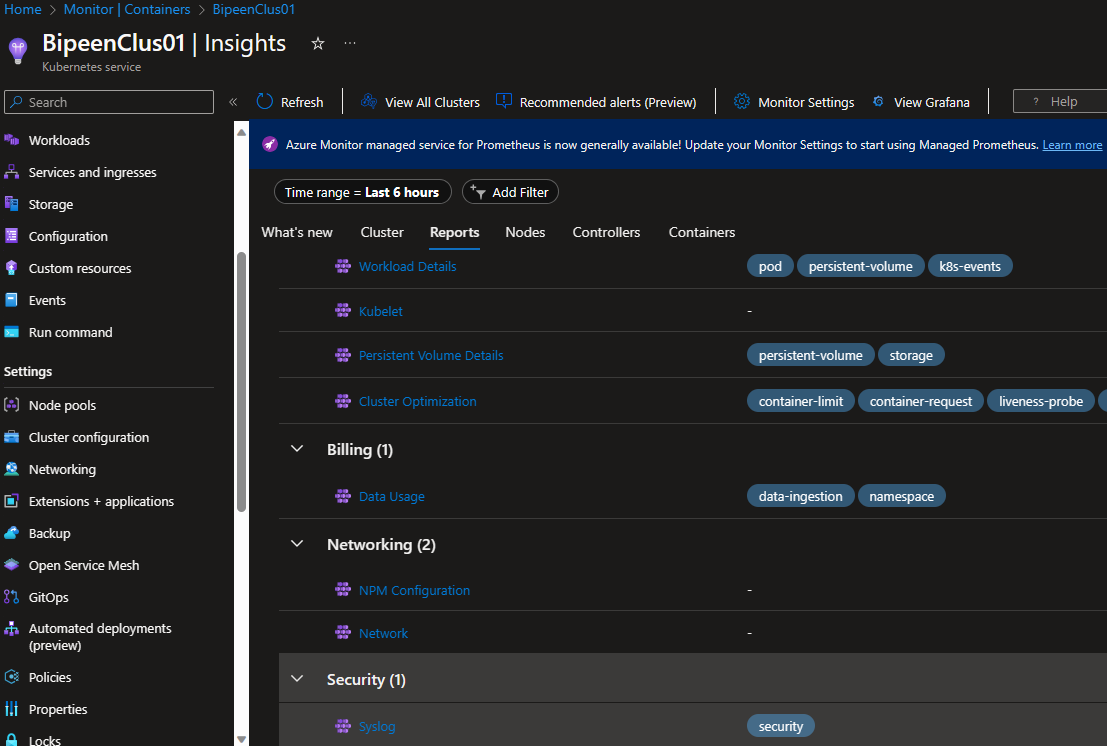
### Access using built-in workbooks

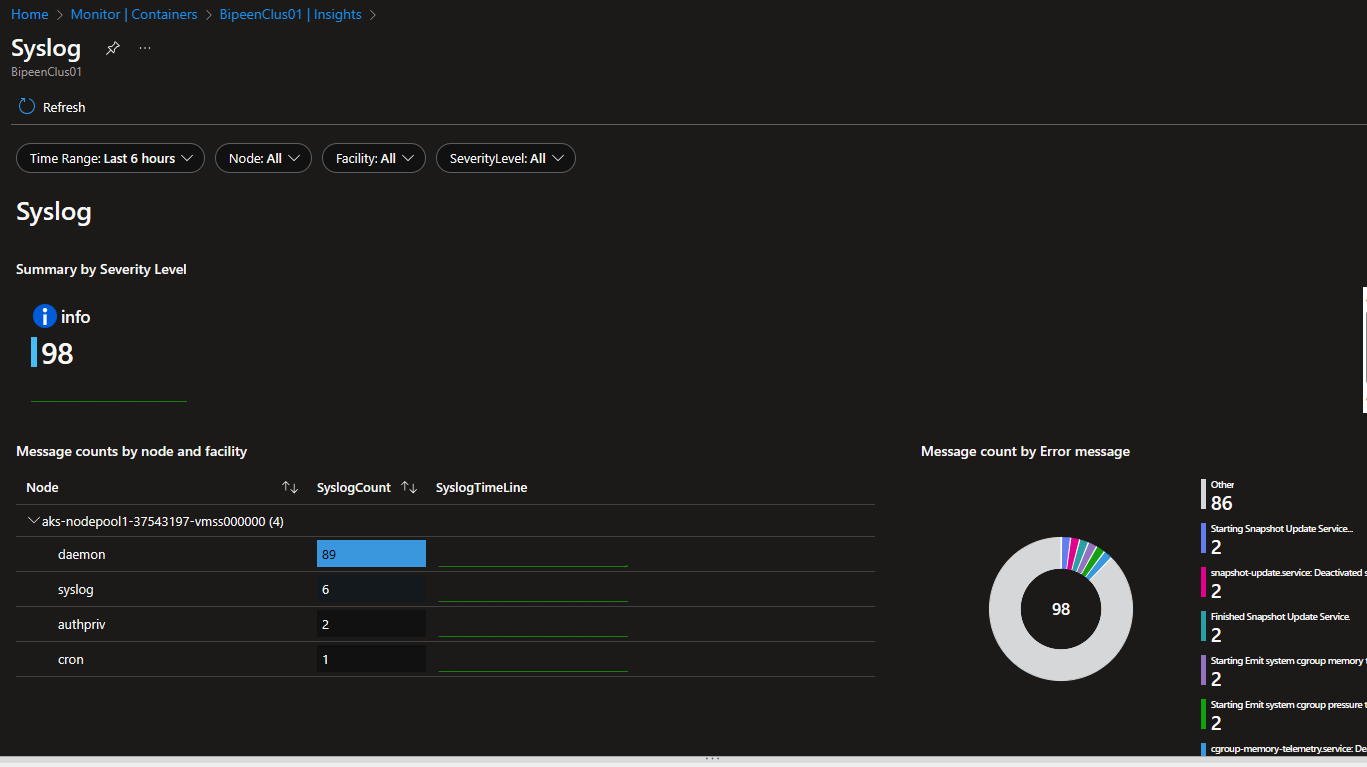
To get a quick snapshot of your syslog data, customers can use our built-in Syslog workbook. There are two ways to access the built-in workbook.

Option 1 - The Reports tab in Container Insights. Navigate to your cluster. Open the Insights tab for your cluster. Open the Reports tab and look for the Syslog workbook.

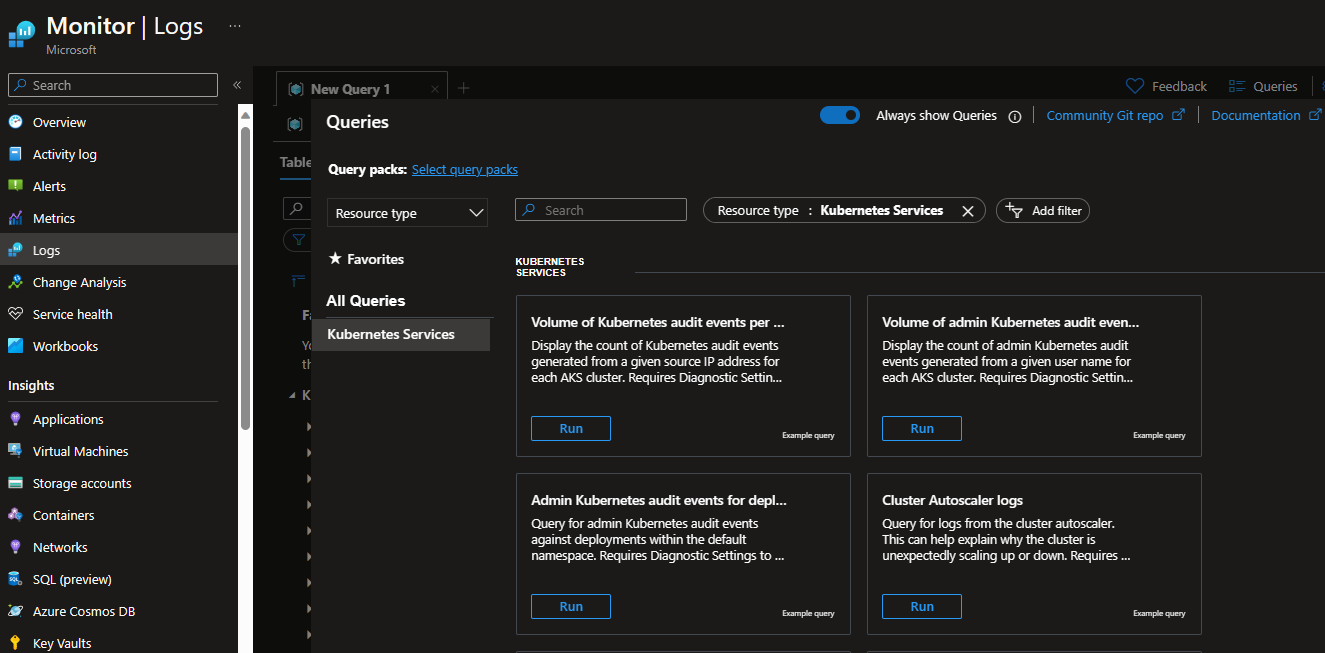


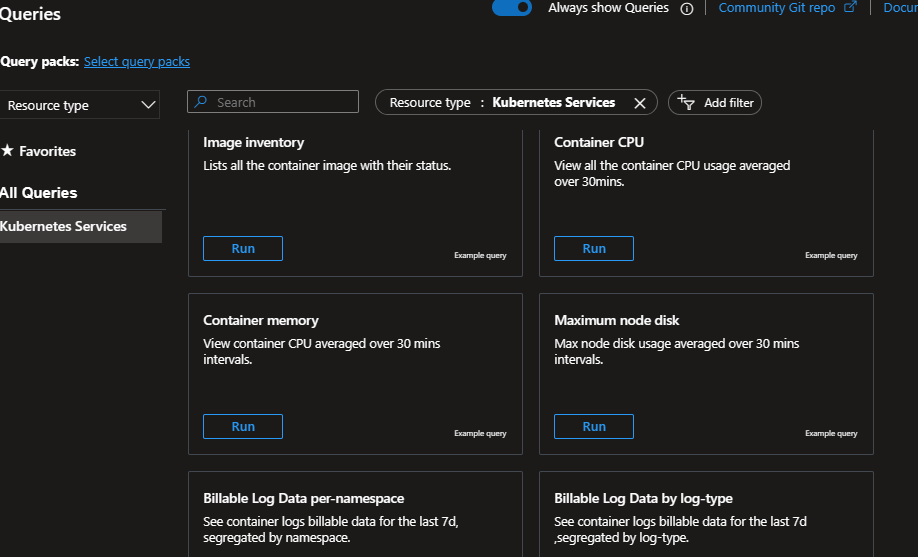
Option 2 - The Workbooks tab in AKS Navigate to your cluster. Open the Workbooks tab for your cluster and look for the Syslog workbook.

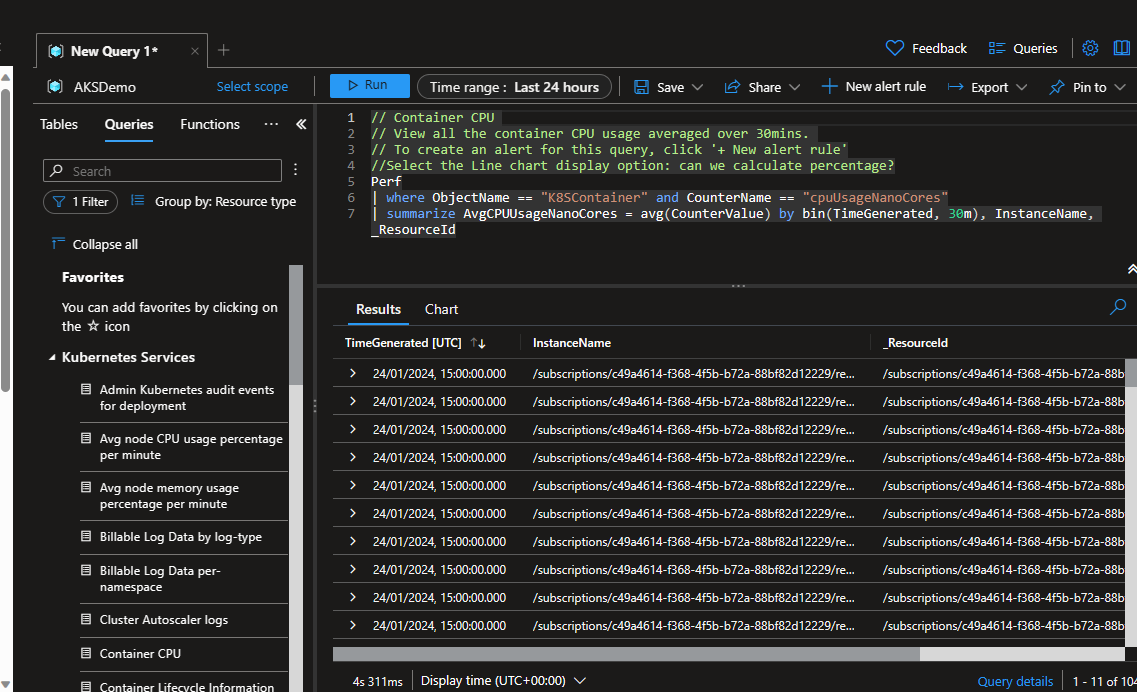




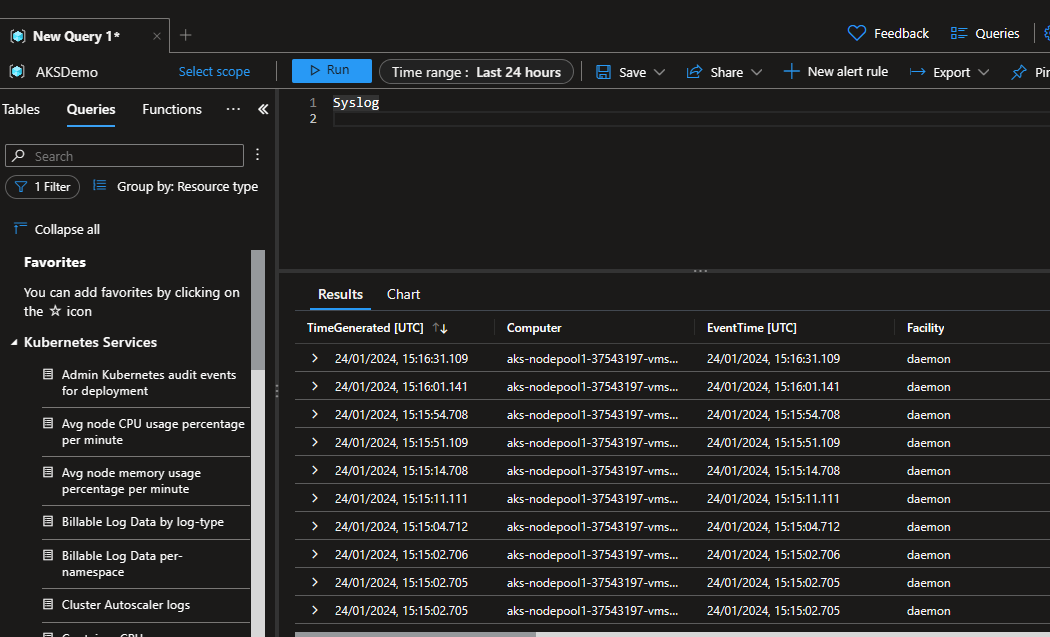
Now Log the quesry





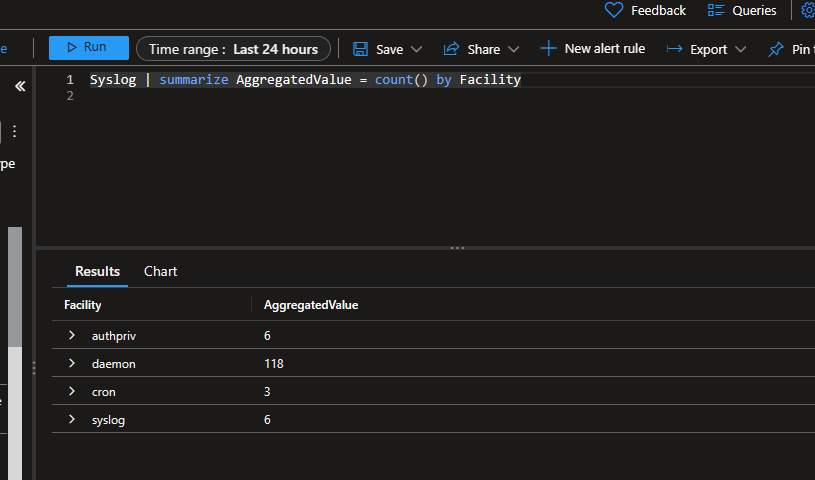


Only type syslogs give all logs



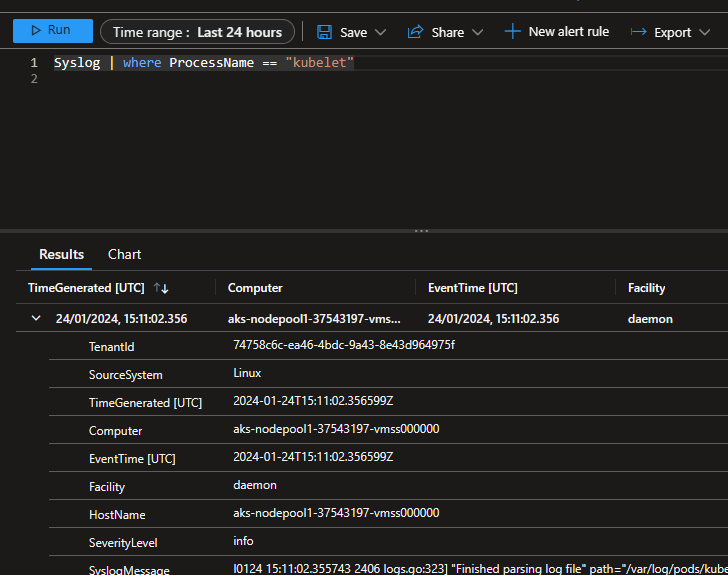
Count of Syslog records by facility

Syslog | summarize AggregatedValue = count() by Facility



All Syslog records from the kubelet process

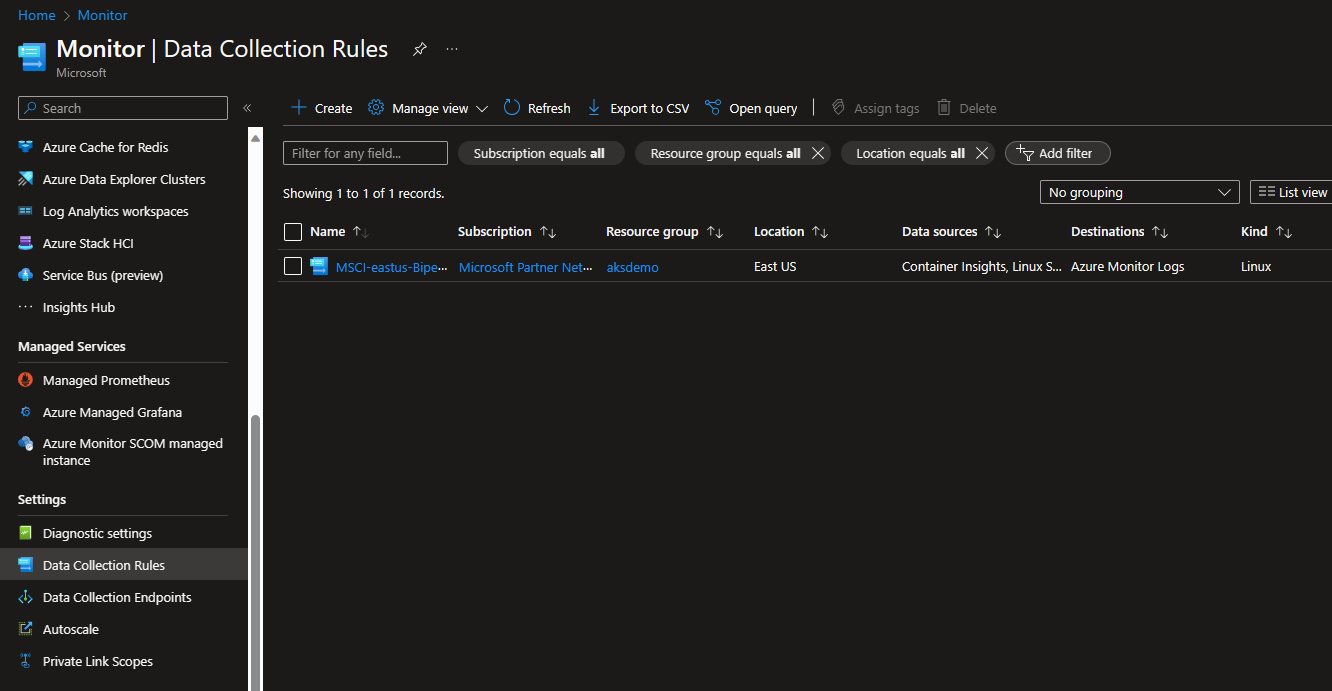
Syslog | where ProcessName == "kubelet"



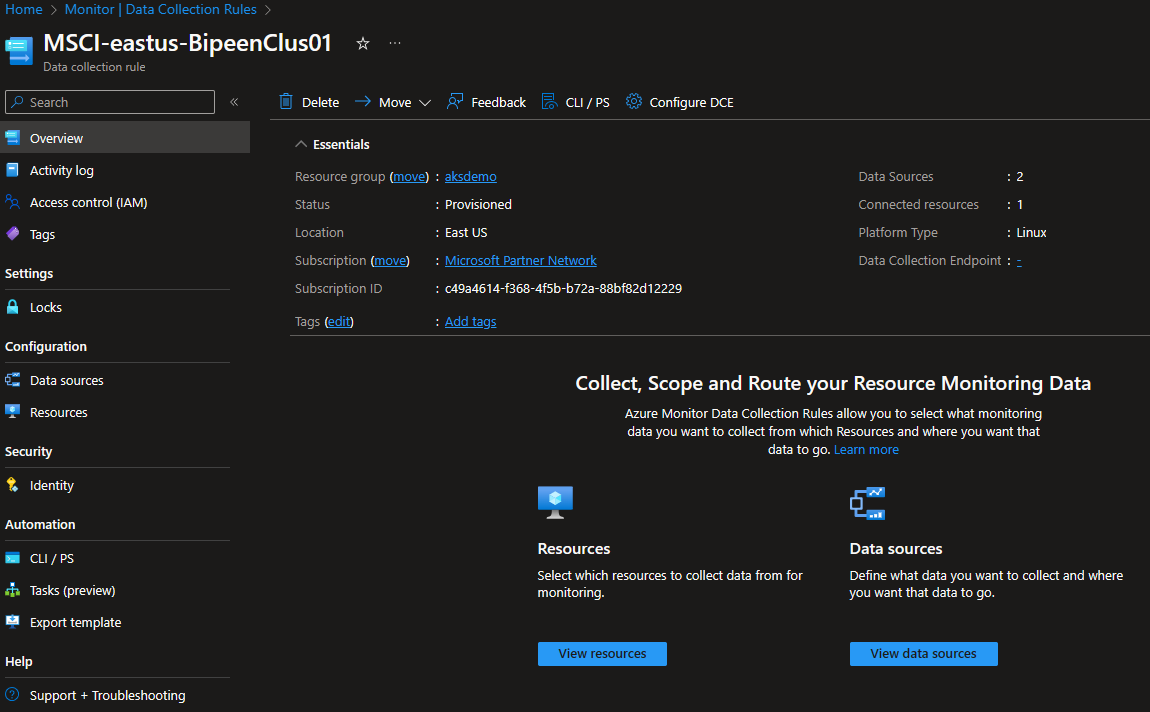
## Editing your Syslog collection settings

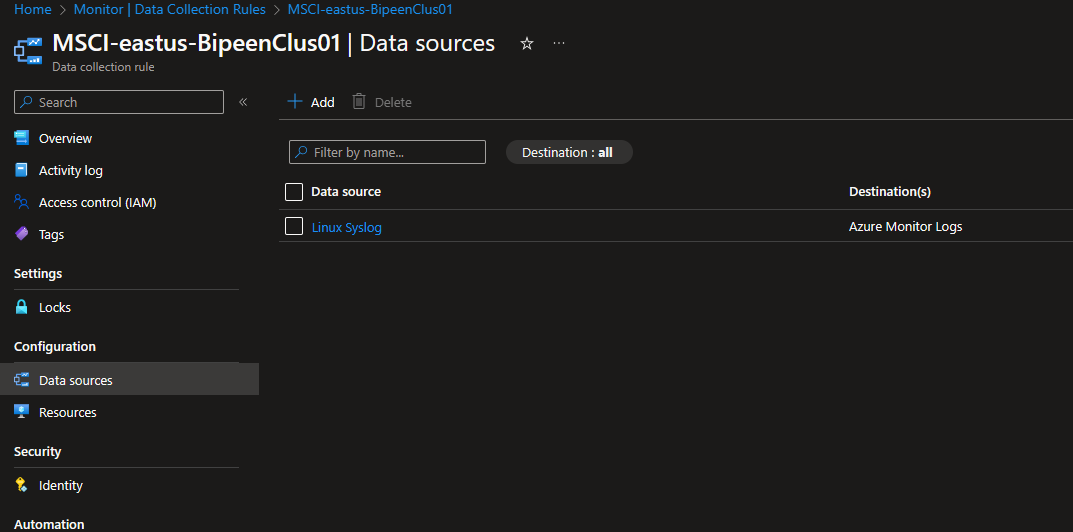
To modify the configuration for your Syslog collection, you modify the [data collection rule (DCR)](https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/data-collection-rule-overview) that was created when you enabled it.

Select **Data Collection Rules** from the **Monitor** menu in the Azure portal.

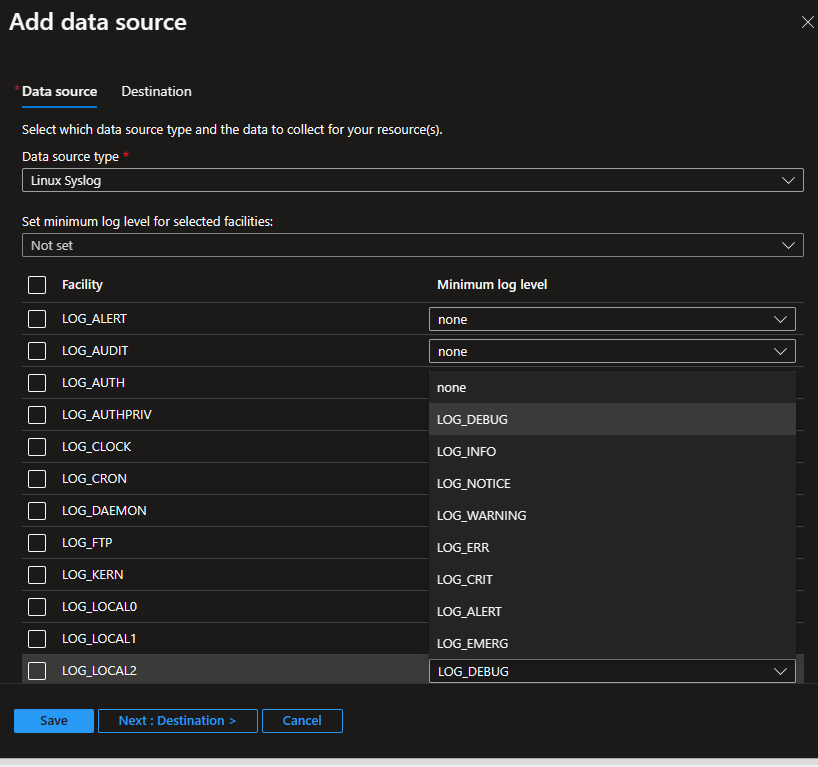


Select your DCR and then **View data sources**. Select the **Linux Syslog** data source to view the Syslog collection details





Select the minimum log level for each facility that you want to collect.



Eg as below

