Docs / Learn / Browse / Azure Kubernetes Service Workshop / Exercise - Configure monitoring for your application

< Previous

Next >

Copy

Sign in

□ Bookmark

Unit 9 of 11 ∨

Exercise - Configure monitoring for your 100 XP application 10 minutes The success of Fruit Smoothies' marketing campaign is the ongoing performance of the ratings website. The performance is depended on your cluster's performance and relies on the fact that you can monitor the different components in your application, view logs, and get alerts whenever your application goes down or some parts of it fail. You can use a combination of available tools to set up alerting capabilities for your application. In this exercise, you will: ✓ Create a Log Analytics workspace ✓ Enable the AKS monitoring add-on ✓ Inspect the AKS event logs and monitor cluster health ✓ Configure Kubernetes RBAC to enable live log data ✓ View the live container logs and AKS events Create a Log Analytics workspace Azure Monitor for containers is a comprehensive monitoring solution for Azure Kubernetes Service. This solution gives you insight into the performance of your cluster by collecting memory and processor metrics from controllers, nodes, and containers. You use Log Analytics in Azure Monitor to store monitoring data, events, and metrics from your AKS cluster and the applications. First, you'll pre-create the Log Analytics workspace in your assigned environment resource group. 1. Sign in to Azure Cloud Shell with an Azure account.

- 2. You need a unique name for the workspace. Run the command below to generate a name similar to aksworkshop-
- Copy bash WORKSPACE=aksworkshop-workspace-\$RANDOM

3. Run the az resource create command to create the workspace in the same resource group and region as your Azure Kubernetes Service (AKS) cluster. For example, aksworkshop in East US. 🗅 Сору bash az resource create --resource-type Microsoft.OperationalInsights/workspaces \ --name \$WORKSPACE \ --resource-group \$RESOURCE_GROUP \ --location \$REGION_NAME \ --properties '{}' -o table

Once the workspace is ready, you can integrate the Azure Monitor add-on and enable container monitoring on your AKS

--name \$AKS_CLUSTER_NAME \

--workspace-resource-id \$WORKSPACE_ID

--addons monitoring \

! Note

Enable the AKS monitoring add-on

workspace-12345.

cluster. 1. You need to provide the resource ID of your workspace to enable the add-on. Run the following command to

Azure CLI

retrieve and store the workspace ID in a Bash variable named WORKSPACE_ID.

WORKSPACE_ID=\$(az resource show --resource-type Microsoft.OperationalInsights/workspaces \ --resource-group \$RESOURCE_GROUP \ --name \$WORKSPACE \ --query "id" -o tsv) 2. Next, enable the monitoring add-on by running the az aks enable-addons command. **С**ору bash az aks enable-addons \ --resource-group \$RESOURCE_GROUP \

minutes for data to appear for your cluster.

Inspect the AKS event logs and monitor cluster health

you a global perspective of all containers deployed across subscriptions and resource groups. From here, you can track containers that are monitored and those containers that aren't monitored. You can also inspect each container's statistics

individually. Let's look at the steps you need to take to get a detailed view of the health of nodes and pods in a cluster. 1. Switch to the <u>Azure portal</u> .

4. Select the Cluster tab at the top of the view to check the cluster utilization. Notice how this view is again a high-level

We view utilization reports and charts for your cluster in the Azure portal by using Azure Monitor. Azure Monitor gives

It might take some time to establish monitoring data flow for newly created clusters. Allow at least 5 to 10

2. Select **Azure Monitor** from the left pane in the Azure portal.

3. Under the **Insights** section, select **Containers** to see a list of all clusters that you have access to.

view that provides you a view on the cluster, nodes, controllers, and containers.

What's new Cluster Health (Preview) Nodes Controllers Containers Deployments (Preview)

- Node memory utilization % Avg Min 50th 90th 95th Max 🖈 Node count Active pod count Total Ready Not Ready

Total Pending Running Unknown Succeeded Failed 🕉

doing. What's new Cluster Health (Preview) Nodes Controllers Containers Deployments (Preview) Kubernetes infrastructure Current state Kubernetes API server

5. Select the **Health** tab at the top of the view to get a view on how the AKS infrastructure services of the cluster are

 azure-cni-networkmonitor (DaemonSet) Last recalculated a few seconds ago on January 22, 2020 11:58 AV azure-ip-masq-agent (DaemonSet) 17 minutes ago on January 22, 2020 11:42 AM coredns-autoscaler (ReplicaSet) kube-proxy (DaemonSet) HEALTH ASPECT STATE kubernetes-dashboard (ReplicaSet) Healthy Kubernetes infrastructure metrics-server (ReplicaSet) omsagent (DaemonSet) Healthy omsagent-rs (ReplicaSet) Unnelfront (ReplicaSet) 6. Select the **Nodes** tab at the top of the view to get a detailed view of your nodes' health and pods in the cluster. What's new Cluster Health (Preview) Nodes Controllers Containers Deployments (Preview) Search by name... TREND 95TH % (1 BAR = 1M)

■ cert-manager-web...
✓ Ok cert-manager-webh. cert-manager-webh. tunnelfront-857f55... 6 days

tunnelfront-857f55

6 days

To enable and set permissions for the agent to collect the data, first, create a *Role* that has access to pod logs and events. Then you'll assign permissions to users by using RoleBinding.

In addition to the high-level overview of your cluster's health, you can also view live log data of specific containers.

Configure Kubernetes RBAC to enable live log data

We use role-based access control (RBAC) in Kubernetes as a way of regulating access to resources based on the roles of individual users within your organization. RBAC authorization uses a set of related paths in the Kubernetes API to allow you to dynamically configure policies. The RBAC API defines four Kubernetes objects:

Role ClusterRole RoleBinding ClusterRoleBinding

What is role-based access control (RBAC)?

What is a Kubernetes Role? The RBAC Role and ClusterRole objects allow you to set up rules that represent a set of permissions. The main difference

given namespace.

bash

2. Paste the following text in the file.

kind: ClusterRole

between a Role and a ClusterRole is that a Role is used with resources in a specific namespace and ClusterRole is used with non-namespace resources in a cluster. You'll see how to define a ClusterRole later in the exercise.

What is a Kubernetes RoleBinding? We use a role binding to grant the permissions defined in a role to a user or set of users. A role binding contains the list

of users, groups, or service accounts, and a reference to the role being granted. Like the Role and ClusterRole, a RoleBinding grants permission within a specific namespace and the ClusterRoleBinding grants access to the cluster. You'll use a ClusterRoleBinding bind your ClusterRole to all the namespaces in your cluster. In this exercise, you'll set up *Roles* and *RoleBindings* that aren't limited to a specific namespace. You can configure *Roles*

and RoleBindings to grant permissions and bind roles to users across the entire cluster or to cluster resources outside a

code logreader-rbac.yaml

YAML apiVersion: rbac.authorization.k8s.io/v1

1. Create a file called logreader-rbac.yaml by using the integrated editor.

- metadata: name: containerHealth-log-reader rules: - apiGroups: ["", "metrics.k8s.io", "extensions", "apps"] resources: - "pods/log" - "events" - "nodes" - "pods" - "deployments" - "replicasets" verbs: ["get", "list"] apiVersion: rbac.authorization.k8s.io/v1 kind: ClusterRoleBinding metadata: name: containerHealth-read-logs-global roleRef: kind: ClusterRole name: containerHealth-log-reader apiGroup: rbac.authorization.k8s.io subjects: - kind: User name: clusterUser apiGroup: rbac.authorization.k8s.io 3. To save the file, select Ctrl+S. To close the editor, select Ctrl+Q. 4. Apply the configuration by using the kubectl apply command. Copy bash
- View the live container logs and AKS events 1. Switch back to the AKS cluster in the Azure portal.
 - ratings-api container. The new view allows you to debug the status of the container. What's new Cluster Health (Preview) Nodes Controllers Containers Deployments (Preview) Search by name..

2. Select **Insights** under **Monitoring**.

kubectl apply \

-f logreader-rbac.yaml

- △ Scroll || Pause 🗵 Clear **Container Status** 20-01-22T20:16:16.223471565Z '}'

3. Select the **Controllers** tab, and choose a container to view its live logs or event logs. For example, choose the

Container ID Pod name: ratings-api-564446d9c4-xwdfk (ratings-api) d7bba2e38ab41e07ef9a054fe315ffffb9aaeb1ace31c11 (1 New Logs, 0 Event(s) found) 🔮 20-01-22T20:16:16.223741966Z 'Saving rating{\n' + 0-01-22T20:16:16.223753466Z " raterlp: '127.0.0.1',\n" + 20-01-22T20:16:16.223757466Z '_id: 5e28ad908db8770018898516,\n' + ratings-api 20-01-22T20:16:16.223760666Z ' itemRated: [5e13aac58db8770018898511],\n' + 20-01-22T20:16:16.223763866Z 'timestamp: 2020-01-22T20:16:16.223Z,\n'+ Image Tag 20-01-22T20:16:16.223766966Z 'rating: 0\n' + 20-01-22T20:16:16.223770066Z '}' Summary In this exercise, you created a Log Analytics workspace in Azure Monitor to store monitoring and logging data for your AKS cluster. You enabled the AKS monitoring add-on to enable the collection of data, and inspected the AKS cluster

health. You then used Kubernetes RBAC to enable the collection of live logging data and then viewed live log data in the

Azure portal.

Next, we'll take a look at scaling the Fruit Smoothies AKS cluster. Next unit: Exercise - Scale your application to meet demand

Continue >

Need help? See our troubleshooting guide or provide specific feedback by reporting an issue.

ratings-api
Container

☑ View in analytics

Container Name ratings-api

TREND 95TH % (1 BAR = 1M)

View live data (preview)

Copy

🖺 Сору