

#Setup ADO agentpool with Autoscale mode under K8s cluster.

1. Setting up AKS

Microsoft Azure

Search resources, services, and docs (G+)

Home > Kubernetes services >

Kubernetes services

pwc (PwC.onmicrosoft.com)

+ Create Manage view

Filter for any field...

Name ↑

Type ↑

poc

Kubernetes service

Create Kubernetes cluster

Basics Node pools Authentication Networking Integrations Tags Review + create

Azure Kubernetes Service (AKS) manages your hosted Kubernetes environment, making it quick and easy to deploy and manage containerized applications without container orchestration expertise. It also eliminates the burden of ongoing operations and maintenance by provisioning, upgrading, and scaling resources on demand, without taking your applications offline. [Learn more about Azure Kubernetes Service](#)

Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Visual Studio Enterprise Subscription

Resource group *

(New) Resource group

Create new

Cluster details

Preset configuration

Standard

Quickly customize your cluster by choosing the preset configuration applicable to your scenario. Depending on the selection, values of certain fields might change in different tabs. You can modify these values at any time.

[View all preset configurations](#)

Review + create

< Previous

Next : Node pools >

Primary node pool

The number and size of nodes in the primary node pool in your cluster. For production workloads, at least 3 nodes are recommended for resiliency. For development or test workloads, only one node is required. If you would like to add additional node pools or to see additional configuration options for this node pool, go to the 'Node pools' tab above. You will be able to add additional node pools after creating your cluster. [Learn more about node pools in Azure Kubernetes Service](#)

Node size *

Standard DS2 v2

Standard DS2_v2 is recommended for standard configuration.

[Change size](#)

Scale method *

Manual

Autoscale

Autoscaling is recommended for standard configuration.

Node count *

3

Review + create

< Previous

Next : Node pools >

Home > Kubernetes services > Create Kubernetes cluster >

Select a VM size

Search by VM size...

Display cost: Monthly

vCPUs: All

RAM (GiB): All

Add filter

Showing 315 VM sizes. | Subscription: Visual Studio Enterprise Subscription | Region: East US | Current size: Standard_DS2_v2 | [Learn more about VM sizes](#) | Group by series

VM Size	Family	vCPUs	RAM (GiB)	Data disks	Max IOPS	Temp storage (GiB)	Premium disk	Cost/month
Most used by Azure users								
The most used sizes by users in Azure								
D51_v2	General purpose	1	3.5	4	3200	7	Supported	Loading...
D2s_v3	General purpose	2	8	4	3200	16	Supported	Loading...
D2as_v4	General purpose	2	8	4	3200	16	Supported	Loading...
B2s	General purpose	2	4	4	1280	8	Supported	Loading...
B2ms	General purpose	2	8	4	1920	16	Supported	Loading...
DS2_v2	General purpose	2	7	8	6400	14	Supported	Loading...
B4ms	General purpose	4	16	8	2880	32	Supported	Loading...
D4s_v3	General purpose	4	16	8	6400	32	Supported	Loading...
DS3_v2	General purpose	4	14	16	12800	28	Supported	Loading...
D8s_v3	General purpose	8	32	16	12800	64	Supported	Loading...

Select

Create Kubernetes cluster ...

Basics **Node pools** Authentication Networking Integrations Tags Review + create

Node pools

In addition to the required primary node pool configured on the Basics tab, you can also add optional node pools to handle a variety of workloads [Learn more about node pools](#)

+ Add node pool  Delete

Name	Mode	OS type	Node count	Node size
<input type="checkbox"/> agentpool	System	Linux	1	Standard_D2s_v3

Enable virtual nodes

Virtual nodes allow burstable scaling backed by serverless Azure Container Instances. [Learn more about virtual nodes](#)

Enable virtual nodes  ☐

Enable virtual machine scale sets

Enabling virtual machine scale sets will create a cluster that uses virtual machine scale sets instead of individual virtual machines for the cluster nodes. Virtual machine scale sets are required for scenarios including autoscaling, multiple node pools, and Windows support. [Learn more about virtual machine scale sets in AKS](#)

Review + create


< Previous

Next : Authentication >

Add a node pool ...

OS type * 

- ☒ Linux
☐ Windows

Availability zones 


None


Node size * 

Standard DS1 v2
1 vcpu, 3 GiB memory
[Choose a size](#)

Scale method * 

- ☒ Manual
☐ Autoscale

Node count * 

Max pods per node * 

110 
10 - 250

Add

Cancel

Create Kubernetes cluster ...

Basics Node pools **Authentication** Networking Integrations Tags Review + create

Cluster infrastructure

The cluster infrastructure authentication specified is used by Azure Kubernetes Service to manage cloud resources attached to the cluster. This can be either a [service principal](#) or a [system-assigned managed identity](#).

Authentication method ☐ Service principal ☒ System-assigned managed identity

Kubernetes authentication and authorization

Authentication and authorization are used by the Kubernetes cluster to control user access to the cluster as well as what the user may do once authenticated. [Learn more about Kubernetes authentication](#)

Role-based access control (RBAC) ☒ Enabled ☐ Disabled

AKS-managed Azure Active Directory ☐

Node pool OS disk encryption

By default, all disks in AKS are encrypted at rest with Microsoft-managed keys. For additional control over encryption, you can supply your own keys using a disk encryption set backed by an Azure Key Vault. The disk encryption set will be used to encrypt the OS disks for all node pools in the cluster. [Learn more](#)

Encryption type (Default) Encryption at-rest with a platform-managed key

Review + create

< Previous

Next : Networking >

Create Kubernetes cluster ...

Application pods are connected directly to the VNet, which allows for native integration with VNet features.

[Learn more about networking in Azure Kubernetes Service](#)

Network configuration ☐ Kubenet ☒ Azure CNI
i The Azure CNI plugin requires an IP address from the subnet below for each pod on a node, which can more quickly exhaust available IP addresses if a high value is set for pods per node. Consider modifying the default values for pods per node for each node pool on the "Node pools" tab. [Learn more](#)

Virtual network *

Cluster subnet *

Kubernetes service address range *

Kubernetes DNS service IP address *

Docker Bridge address *

DNS name prefix *


Traffic routing


Review + create



< Previous



Next : Integrations >

Create Kubernetes cluster ...


Kubernetes service address range 


Kubernetes DNS service IP address * 

Docker Bridge address *  


DNS name prefix *  


Traffic routing


Load balancer  Standard

Enable HTTP application routing  ☒

Security

Enable private cluster  ☐

Set authorized IP ranges  ☐

Network policy  ☒ None
☐ Calico
☐ Azure


Create Kubernetes cluster ...

Basics Node pools Authentication Networking Integrations Tags Review + create

Connect your AKS cluster with additional services.

Azure Container Registry

Connect your cluster to an Azure Container Registry to enable seamless deployments from a private image registry. You can create a new registry or choose one you already have. [Learn more about Azure Container Registry](#) 

Container registry 
[Create new](#)

Azure Monitor

In addition to the CPU and memory metrics included in AKS by default, you can enable Container Insights for more comprehensive data on the overall performance and health of your cluster. Billing is based on data ingestion and retention settings.

[Learn more about container performance and health monitoring](#)

[Learn more about pricing](#)

Container monitoring ☒ Enabled ☐ Disabled
 Azure monitor is recommended for standard configuration.

Log Analytics workspace  
[Create new](#)

Azure Policies

[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

Home > Kubernetes services >

Create Kubernetes cluster ...

Running final validation...

Basics Node pools Authentication Networking Integrations Tags Review + create

Basics

Subscription	Visual Studio Enterprise Subscription
Resource group	(new) test_group
Region	East US
Kubernetes cluster name	test
Kubernetes version	1.20.9

Node pools

Node pools	2
Enable virtual nodes	Disabled
Enable virtual machine scale sets	Enabled

Authentication

Authentication method	System-assigned managed identity
-----------------------	----------------------------------

Create < Previous Next > Download a template for automation

Microsoft Azure Search resources, services, and docs (G+)

Home > Kubernetes services > Create Kubernetes cluster >

Template ...

Download Add to library (preview) Deploy Visualize template

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. [Learn more about temp](#)

☒ Include parameters ⓘ

Template Parameters Scripts

Parameters (19)

Variables (0)

Resources (4)

- [parameters('resourceName')]
(Microsoft.ContainerService/mana
- ConnectAKStoACR-b7d3b4f1-
1d9d-47cb-9463-0e10194c404f
(Microsoft.Resources/deployment
- test_group-vnet
(Microsoft.Network/virtualNetworl
- ClusterSubnetRoleAssignmentDep
20210929121441
(Microsoft.Resources/deployment

```
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "resourceName": {
6       "type": "string",
7       "metadata": {
8         "description": "The name of the Managed Cluster resource."
9       }
10    },
11    "location": {
12      "type": "string",
13      "metadata": {
14        "description": "The location of AKS resource."
15      }
16    },
17    "dnsPrefix": {
18      "type": "string",
19      "metadata": {
```

ARM template for this cluster config.



aks_arm_template.zip

Microsoft Azure

Home > poc

Kubernetes service

Search (Ctrl+/)

Connect Start Stop Delete Refresh Give feedback

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Security

Kubernetes resources

Namespaces

Workloads

Services and ingresses

Storage

Configuration

Settings

Node pools

https://portal.azure.com/#on

Essentials

Resource group (change) : poc_group

Status : Succeeded (Running)

Location : East US

Subscription (change) : Visual Studio Enterprise Subscription

Subscription ID : 721cf6b6-a86a-442c-ab09-756cc37d0a2a

Tags (change) : [Click here to add tags](#)

Get started Properties Monitoring Capabilities Recommendations (5) Tutorials

Kubernetes services

Encryption type : Encryption at-rest with a platform-managed key

Virtual node pools : Not enabled

Node pools

Node pools : 1 node pool

Kubernetes versions : 1.20.9

Node sizes : Standard_DS2_v2

Connect to poc

Connect to your cluster using command line tooling to interact directly with cluster using kubectl, the command line tool for Kubernetes. Kubectl is available within the Azure Cloud Shell by default and can also be installed locally. [Learn more](#)

1. Open Cloud Shell or the Azure CLI

2. Run the following commands

az account set --subscription 721cf6b6-a86a-442c-ab09-756cc37d0a2a

az aks get-credentials --resource-group poc_group --name poc

Sample commands

Once you have run the command above to connect to the cluster, you can run any kubectl commands. Here are a few examples of useful commands you can try.

List all deployments in all namespaces

kubectl get deployments --all-namespaces=true

List all deployments in a specific namespace

Format:kubectl get deployments --namespace <namespace-name>

kubectl get deployments --namespace kube-system

List details about a specific deployment

Format:kubectl describe deployment <deployment-name> --namespace <namespace-name>

kubectl describe deployment my-dep --namespace kube-system

List pods using a specific label

Format:kubectl get pods -l <label-key>=<label-value> --all-namespaces=true

Log in to Cluster with cloud shell:

```
Bash
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

a @Azure:~$ kubectl get all
NAME                READY   STATUS    RESTARTS   AGE
pod/azdevops-deployment-79b9c8695-141st  1/1     Running   3          14d
pod/vault-agent-injector-686797ccd5-ptxfm  1/1     Running   0          7d21h

NAME                TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
service/external-vault  ClusterIP   10.0.142.169 <none>        8280/TCP   7d20h
service/kubernetes     ClusterIP   10.0.0.1     <none>        443/TCP   28d
service/vault-agent-injector-svc  ClusterIP   10.0.135.53 <none>        443/TCP   7d21h

NAME                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/azdevops-deployment  1/1     1            1          14d
deployment.apps/vault-agent-injector  1/1     1            1          7d21h

NAME                DESIRED   CURRENT   READY   AGE
replicaset.apps/azdevops-deployment-79b9c8695  1         1         1       14d
replicaset.apps/vault-agent-injector-686797ccd5  1         1         1       7d21h

NAME                                                         REFERENCE                                TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
horizontalpodautoscaler.autoscaling/keda-hpa-azure-pipelines-scaledobject  Deployment/azdevops-deployment           0/1 (avg) 1         5         1          21d

a @Azure:~$ ls
clouddrive helm3 sample-app
a @Azure:~$ cd sample-app/
a @Azure:~/sample-app$ ls
'\          deployment2.yaml  dockeragent  get_helm.sh  kaniko      keda.yaml  pod3.yaml  test.sh
config.json deployment.yaml  eks-tf       jobs.yaml   keda2.yaml  pod2.yaml  pod.yaml  working_jfrog.json
```

[Code{manifest file for pod-deployment & autoscore, Dockerfile } for ADO setup on K8s & docker image & ADO build pipeline]



AzureADOAgentCreation-with-k8s.zip

2. ADO agent deployment >>>


```

Bash
service/vault-agent-injector-svc ClusterIP 10.0.135.53 <none> 443/TCP 7d21h

NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/vault-agent-injector 1/1 1 1 7d21h

NAME DESIRED CURRENT READY AGE
replicaset.apps/vault-agent-injector-686797ccd5 1 1 1 7d21h

NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE
horizontalpodautoscaler.autoscaling/keda-hpa-azure-pipelines-scaledobject Deployment/azdevops-deployment 0/1 (avg) 1 5 1 21d

a@h@Azure:~$
a@h@Azure:~$
a@h@Azure:~$
a@h@Azure:~$ kubectl get hpa
NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE
keda-hpa-azure-pipelines-scaledobject Deployment/azdevops-deployment 0/1 (avg) 1 5 1 21d
a@h@Azure:~$ kubectl delete hpa keda-hpa-azure-pipelines-scaledobject
horizontalpodautoscaler.autoscaling "keda-hpa-azure-pipelines-scaledobject" deleted
a@h@Azure:~$ kubectl get all
NAME READY STATUS RESTARTS AGE
pod/vault-agent-injector-686797ccd5-ptxfm 1/1 Running 0 7d21h

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
service/external-vault ClusterIP 10.0.142.169 <none> 8200/TCP 7d20h
service/kubernetes ClusterIP 10.0.0.1 <none> 443/TCP 28d
service/vault-agent-injector-svc ClusterIP 10.0.135.53 <none> 443/TCP 7d21h

NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/vault-agent-injector 1/1 1 1 7d21h

NAME DESIRED CURRENT READY AGE
replicaset.apps/vault-agent-injector-686797ccd5 1 1 1 7d21h
a@h@Azure:~$

a@h@Azure:~$ ls
clouddrive helm3 sample-app
a@h@Azure:~$
a@h@Azure:~$ cd sample-app/
a@h@Azure:~/sample-app$ ls
'\ deployment2.yaml dockeragent get_helm.sh kaniko keda.yaml pod3.yaml test.sh
config.json deployment.yaml eks-tf jobs.yaml keda2.yaml pod2.yaml pod.yaml working_jfrog.json
ashutosh@Azure:~/sample-app$

```

3. Setup PAT token @ADO Organization

The screenshot shows the Azure DevOps interface. On the left, the 'Personal Access Tokens' section is highlighted. The main area displays a list of tokens with columns for Name, Organization, Expiration, and Status. Two tokens are listed: 'aks-agent' and 'kaniko-poc', both with 'Full access' and 'Active' status. A 'New Token' button is visible. On the right, a dropdown menu is open, showing options like 'Preview features', 'Profile', 'Time and Locale', 'Permissions', 'Notifications', 'Theme', 'Usage', 'Personal access tokens', 'SSH public keys', and 'Alternate credentials'. The 'Personal access tokens' option is highlighted.

Take AZP_TOKEN: {{}}

Project Settings
ai-lab

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards
- Boards
- Project configuration
- Team configuration
- Git-Hub connections
- Pipelines
- Agent pools**

Agent pools

Name	Queued jobs	Running jobs
...		
...		
...		1
...		
...		
...		
aks-keda-poc		
...		

Add agent pool

Agent pools are shared across an organization.

Pool to link:

☒ New ☐ Existing

Pool type:

Select an agent pool type

- Self-hosted
- Azure virtual machine scale set

Add agent pool

Agent pools are shared across an organization.

Pool to link:

☒ New ☐ Existing

Pool type:

Self-hosted

A pool of agents that you set up and manage on your own to run jobs. [Learn more.](#)

Name:

dddd

Description (optional):

[Markdown supported.](#)

Pipeline permissions:

☒ Grant access permission to all pipelines

Create

```
linux-agents > deployments > ! agent-deploy
1  apiVersion: v1
2  kind: Namespace
3  metadata:
4    name: kaniko
5  ---
6  apiVersion: v1
7  kind: Secret
8  metadata:
9    name: kanikoazdosecret
10   namespace: kaniko
11  data:
12    AZP_TOKEN: __azp-token__
13  ---
14  apiVersion: apps/v1
15  kind: Deployment
16  metadata:
17    name: agent-deployment
18    namespace: kaniko
19    labels:
20      name: agent-deployment
21  spec:
22    replicas: 1
23    selector:
24      matchLabels:
25        app: agent-deployment
26    template:
27      metadata:
28        labels:
29          app: agent-deployment
```

```
Bash
app: azdevops-agent
name: azdevops-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: azdevops-agent
  template:
    metadata:
      labels:
        app: azdevops-agent
    spec:
      containers:
        - name: azdevops-agent
          image: kanikopoc.azurecr.io/samplekaniko:11544
          env:
            - name: AZP_URL
              value: "https://dev.azure.com/!-g-ad-ory"
            - name: AZP_POOL
              value: "aks-keda-poc"
            - name: AZP_TOKEN
              valueFrom:
                secretKeyRef:
                  name: azdevops
                  key: AZP_TOKEN
          volumeMounts:
            - mountPath: /var/run/docker.sock
              name: docker-volume
          volumes:
            - name: docker-volume
              hostPath:
                path: /var/run/docker.sock
...@Azure:~/sample-app$ kubectl apply -f deployment.yaml
```



```

Bash
ashutosh@Azure:~/sample-app$ cat keda.yaml
apiVersion: v1
kind: Secret
metadata:
  name: pipeline-auth
data:
  personalAccessToken: -----
# organizationURL: # -----
---
apiVersion: keda.sh/v1alpha1
kind: TriggerAuthentication
metadata:
  name: pipeline-trigger-auth
spec:
  secretTargetRef:
    - parameter: personalAccessToken
      name: pipeline-auth
      key: personalAccessToken
    - parameter: organizationURL
      name: pipeline-auth
      key: organizationURL
---
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: azure-pipelines-scaledobject
spec:
  scaleTargetRef:
    name: azdevops-deployment
# app: azdevops-agent
minReplicaCount: 1
maxReplicaCount: 5
triggers:

```

```

Bash
apiVersion: keda.sh/v1alpha1
kind: TriggerAuthentication
metadata:
  name: pipeline-trigger-auth
spec:
  secretTargetRef:
    - parameter: personalAccessToken
      name: pipeline-auth
      key: personalAccessToken
    - parameter: organizationURL
      name: pipeline-auth
      key: organizationURL
---
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: azure-pipelines-scaledobject
spec:
  scaleTargetRef:
    name: azdevops-deployment
# app: azdevops-agent
minReplicaCount: 1
maxReplicaCount: 5
triggers:
  - type: azure-pipelines
    metadata:
      poolID: "1"
      organizationURLFromEnv: AZP_URL
      personalAccessTokenFromEnv: "AZP_TOKEN"
      targetPipelinesQueueLength: "1"
  authenticationRef:
    name: pipeline-trigger-auth

```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
horizontalpodautoscaler.autoscaling/keda-hpa-azure-pipelines-scaledobject	Deployment/azdevops-deployment	0/1 (avg)	1	5	1	12m

```

ashutosh@Azure:~/sample-app$ kubectl apply -f keda.yaml
secret/pipeline-auth unchanged
triggerauthentication.keda.sh/pipeline-trigger-auth unchanged
scaledobject.keda.sh/azure-pipelines-scaledobject unchanged
ashutosh@Azure:~/sample-app$

```

ADO agent Docker file

```
spec:
  containers:
  - name: azdevops-agent
    image: [REDACTED].i/samplekaniko:11544
    env:
      name: AZD_URI
```

2565 Az...

[Agents](#) / [linux-agents](#) / [agent-capabilities](#) / Dockerfile

 **██████████** modified a few things

Contributors

52 lines (47 sloc) | 1.78 KB

```

1 FROM ubuntu:18.04
2
3 # To make it easier to build and release pipelines to run apt-get,
4 # configure apt to not require confirmation (assume the -y argument by default)
5 ENV DEBIAN_FRONTEND=noninteractive
6 ARG JFROG_PASSWORD
7 ARG JFROG_USER
8 ENV JFROG_USER=$JFROG_USER
9 ENV AILAB_JFROG_PASSWORD=$JFROG_PASSWORD
10 RUN echo "APT::Get::Assume-Yes \"true\";" > /etc/apt/apt.conf.d/90assumeyes
11
12 ## Microsoft suggested capabilities for Agent setup ##
13 RUN apt-get update && apt-get install -y --no-install-recommends \
14     ca-certificates \
15     curl \
16     jq \
17     git \
18     iputils-ping \
19     libcurl4 \
20     libc6 \
21     libunwind8 \

```

Delete Auto scaled offline agents from agents' pool:

update PAT token @agents_delete.py:

```
# REST API auth connection
def api_connection():
    pat = 'xxxxxxxxxxxxxxxxii2ujyxpmmzsrchrvi42xxxxxxxxxxxxx'
    authorization = str(base64.b64encode(bytes(':'+pat, 'ascii')), 'ascii')
```

azure-pipelines.yml >>> update task in pipeline

```

102 inputs:
103   code: |
104     kubectl apply -f ./linux-agents/deployments/scaling.yaml
105
106 - task: tsuyoshiushio.shell-exec.shellexec.shellxec@0
107   displayName: 'cleanup offline agent'
108   inputs:
109     code: |
110       python3 ./agents/delete.py

```



agents_delete.py